САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

КАФЕДРА ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ

Камалов Михаил Валерьевич

Магистерская диссертация

Разработка поисковой системы для базы данных MEDLINE выполняющей ранжирование результатов поиска с позиции доказательной медицины и системы критериев GRADE

Направление 010300

Технологии баз данных

Магистерская программа:

Фундаментальная информатика и информационные технологии

Научный руководитель

кандидат физ.-мат. наук, доцент

Добрынин Владимир Юрьевич

Санкт-Петербург

2016

Оглавление

Введение

На данный момент в медицинской практике активно развивается подход, именуемый доказательной медициной [1]. Данный подход требует от специалиста основываться при выборе метода лечения пациента на имеющиеся доказательства достоверности и эффективности рассматриваемого метода. Сложность применения доказательной медицины на практике заключается в оценке уровня качества и надежности уже существующих медицинских исследований. Для оценки качества медицинского исследования в доказательной медицине используется шкала, ранжирующая исследования по уровню доказательности.

Для удобного описания в дальнейшем уровней доказательности введем понятие *рандомизированное контролируемое исследование (РКИ)*. Данное понятие описывает медицинское исследование, выполненное в соответствии с рядом требований:

1. в процессе проведения исследования формирование групп пациентов происходило случайным образом (рандомизированно);

2. в процессе проведения исследования применялись техники ослепления.

Под техникой ослепления понимается процесс, при котором пациент (группа пациентов) не знает о медицинском вмешательстве, которое получает в процессе проведения эксперимента. Под медицинским вмешательством понимается выбор подхода при лечении пациента. Стандартным примером РКИ является исследование, в котором медицинским вмешательством является разрабатываемый лекарственный препарат, а процесс ослепления заключается в том, что одной группе пациентов дают исследуемый препарат, а другой – плацебо, при этом пациенты в группах не знают о том, какой из двух препаратов они принимают. Такой способ проведения эксперимента дает более качественные и надежные результаты эксперимента.

Уровни доказательности

Необходимо заметить, что в разных странах используются разные шкалы уровней доказательности. В данной работе за основу было решено рассматривать шкалу, рекомендованную национальной нормативной документацией[1] США, соответствующую стандартам всемирной организации здравоохранения (ВОЗ). Данная шкала содержит пять уровней доказательности, представленные в табл. 1.

Таблица 1. Уровни доказательности

Уровень	**Тип исследования**
Ia	Данные мета-анализов рандомизированных контролируемых испытаний (РКИ).
Ib	Хотя бы одно РКИ.
IIa	Хотя бы одно хорошо выполненное контролируемое исследование без рандомизации.
IIb	Хотя бы одно хорошо выполненное квази-экспериментальное исследование.
III	Данные из не-экспериментальных описательных исследований, таких как сравнительные исследования, корреляционные исследования.
IV	Экспертное консенсусное мнение либо клинический опыт признанного эксперта.

Поскольку в табл. 1 I и II уровни доказательности имеют подуровни “a” и “b”, то для удобного описания в дальнейшем будем работать не с 5, а с 6 уровнями доказательности, изменив описание уровней из табл. 1 следующим образом: 1 уровень - Ia; 2 уровень - Ib; 3 уровень - IIa; 4 уровень - IIb; 5 уровень - III; 6 уровень - IV.

В соответствии с представленной выше таблицей уровней доказательности рассмотрим более детально каждый из них.

[1] http://www.guideline.gov/

Согласно уровням доказательности табл. 1 мета-анализ занимает первое место, поскольку его задачей является сравнение всех РКИ по определенной тематике. Примером проведения мета-анализа по теме “лечение трахеита у подростков” является сравнение всех РКИ, описывающих применение различных медицинских вмешательств для лечения трахеита у подростков. Ко второму уровню доказательности относятся исследования, проведенные в соответствии с требованиями к РКИ. Третий уровень доказательности занимают исследования, выполненные с частичным удовлетворением требованиям к РКИ, а именно – без выполнения рандомизации при формировании групп пациентов. Четвертому уровню доказательности соответствуют хорошо проведенные квази-экспериментальные исследования. Под квази-экспериментальными исследованиями понимаются исследования, в которых рассматриваются довольно редкие заболевания и набрать значительное число пациентов для выполнения рандомизации и ослепления затруднительно. Например, в случае волчанки [2] можно провести квази-экспериментальное исследование с 3-4 пациентами. К 5-му уровню доказательности относятся все исследования, которые проводят сравнения не РКИ по определенной теме. В приведенном выше примере мета-анализа сравнивались все РКИ по теме “лечение трахеита у подростков”, в случае обзорных (наблюдаемых) исследований будут рассматриваться все не РКИ по данной теме. Частным случаем обзорных исследований являются исследования, описывающие течение какого-либо заболевания без применения медицинского вмешательства. Последний уровень доказательности занимают исследования, качество проведения которых подтверждено не выполнением требований к РКИ, а медицинским экспертом.

Система GRADE

Стоит заметить, что в некоторых случаях исследования, соответствующие первому уровню доказательности, могут содержать ошибки в корректности проведения РКИ. Более подробное описание примеров ошибок в исследованиях с первым уровнем доказательности рассмотрено в [3].

Решением данной проблемы является система Grading of Recommendations Assessment, Development and Evaluation (GRADE)[2]. Данная система оценивает уровень доказательности исследований и ранжирует их по значимости рекомендаций за счет введения дополнительных оценивающих критериев. Дополнительные критерии GRADE (Табл. 2) более подробно рассмотрены в публикации [4]. В GRADE рассматриваются всего два класса рекомендаций: 1-сильные, 2 - слабые, а качество уровня доказательности представлено в 4-х уровнях. Таким образом, используя дополнительные факторы, разрабатываемые рабочей группой GRADE, можно повысить или понизить значимость исследования.

Таблица 2. Критерии оценок GRADE

<table>
<tr><th>Цель исследования</th><th>Качество доказательности</th><th>Понижается если</th><th>Повышается если</th></tr>
<tr><td rowspan="2">Исследования с вмешательством</td><td>Высокое</td><td rowspan="3">Риск отклонения
-1 Серьезный
-2 Очень серьезный
Несовместимость
-1 Серьезная
-2 Очень серьезная
Противоречивость
-1 Серьезная
-2 Очень серьезная
Неточность
-1 Серьезная
-2 Очень серьезная
Предвзятое мнение
-1 Серьезная
-2 Очень серьезная</td><td rowspan="3">Эффективность
+1 Большая
+2 Очень большая
Доля ответа
+1 Свидетельствует о градиенте
Все правдоподобные смешения
+1 Будут повышать демонстрируемый эффект
или
+1 Будут предлагать ложный эффект когда результаты окажутся не эффективными</td></tr>
<tr><td>Умеренное</td></tr>
<tr><td rowspan="2">Наблюдаемые исследования</td><td>Низкое</td></tr>
<tr><td>Очень низкое</td></tr>
</table>

[2] http://www.gradeworkinggroup.org/

Согласно критериям, представленным в табл. 2, можно более детально описать влияние GRADE оценок на уровни доказательности из табл.1. В случае рассмотрения уровней доказательности, медицинское исследование может быть соотнесено к одному из уровней и получит целочисленный ранг (от 1 до 5). Полученный целочисленный ранг в дальнейшем может использоваться в алгоритме ранжирования медицинских исследований по уровням доказательности. В случае рассмотрения GRADE оценок мы добавляем две вещественные весовые переменные к выбранному уровню доказательности, тем самым получая вещественный ранг для уровня доказательности с поправкой по GRADE. Весовые переменные формируются следующим образом:

1. Суммируем оценки, которые можно получить по всем 5 критериям понижения (табл. 2), и делим на максимальную сумму по всем критериям понижения. Обозначим данную переменную как *Decrease*.
2. Суммируем оценки, которые можно получить по всем 3 критериям повышения (табл. 2), и делим на максимальную сумму по всем критериям повышения. Обозначим данную переменную как *Increase*.

Дополнительно введем переменную *Level*, обозначающую уровень доказательности, тогда вещественный ранг для уровня доказательности с поправкой по GRADE обозначим как GRADE_*Score* и будем вычислять следующим образом:

$$\text{GRADE\_}Score = Level + Decrease + Increase$$

На данный момент в работе применяется алгоритм ранжирования, основанный на уровнях доказательности (табл. 1) без учета GRADE критериев. В дальнейшем планируется доработать алгоритм ранжирования с учетом GRADE_*Score*.

Поисковые системы, работающие с базой данных MEDLINE

Необходимо заметить, что на данный момент существует ряд поисковых систем, работающих с крупнейшей библиографической базой исследований по

медицинским наукам MEDLINE[3]. Приведем сравнение трех наиболее известных поисковых систем, а именно: PubMed, TRIP и MEDIE.

***Система* PubMed**

Система PubMed[4] - одна из наиболее активно применяемых на практике поисковых систем. Данная поисковая система обладает рядом возможностей:

1. Полный доступ к MEDLINE, что позволяет выполнять обновление поискового индекса в зависимости от появления новых статей;
2. Формирование поисковой стратегии позволяет указать, по каким именно полям в статье должен выполняться поиск. Пример формирования поисковой стратегии приведен на рис. 1;
3. Дополнительно использует представление медицинских документов концептами, применяя ресурс MeSH[5]. Под концептами понимается список тем, к которым принадлежит данное исследование (аналог списка ключевых слов).

Однако, поисковая стратегия PubMed основана на булевом поиске и не учитывает уровень доказательности медицинских исследований, описываемых в статьях.

[3] http://www.nlm.nih.gov/

[4] http://www.ncbi.nlm.nih.gov/pubmed/

[5] https://www.nlm.nih.gov/mesh/

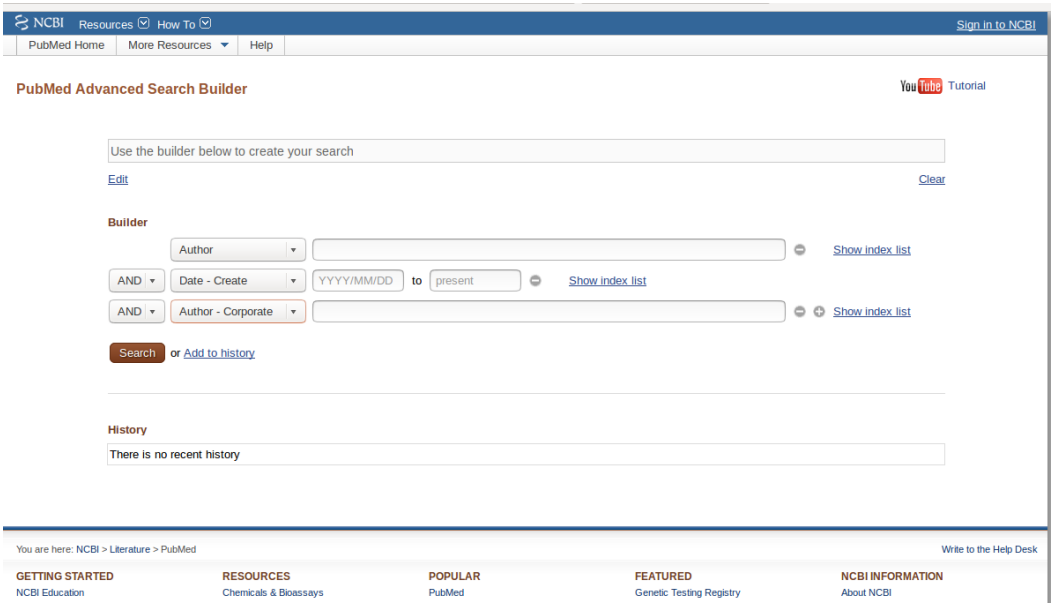

Рис. 1. Формирование поисковой стратегии PubMed.

Система TRIP

Появившаяся в 1997 году поисковая система TRIP[6] [5] также активно применяется на практике и работает с базой данных MEDLINE. Данная система дополнительно индексирует статьи с таких ресурсов как Cochrane[7] и BMJ[8] и обладает следующими возможностями:

1. Имеет богатый набор фильтров, позволяющих значительно локализовать область поиска. Пример фильтров представлен на рис. 2;
2. Алгоритм ранжирования учитывает ресурс, которому принадлежит статья. К примеру, все статьи из Cochrane будут находиться на первом месте по результатам поиска, поскольку данный ресурс содержит только мета-анализы, занимающие первое место по уровням доказательности.

[6] https://www.tripdatabase.com/

[7] http://www.cochrane.org/

[8] http://www.bmj.com

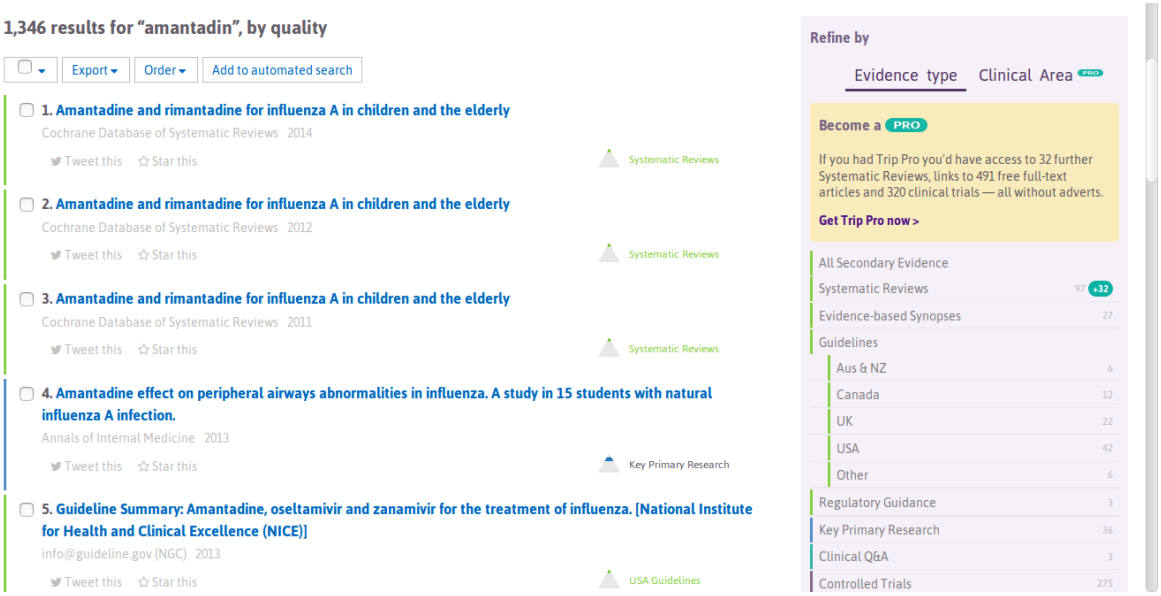

Рис. 2. Примеры фильтров поисковой системы TRIP.

Хотя TRIP и разрабатывалась как медицинская поисковая система с акцентом на доказательную медицину, но алгоритм ранжирования учитывает лишь первый уровень доказательности, основываясь не на содержимом статьи, а на ресурсе, которому принадлежит статья.

Система MEDIE

В 2005 году появилась новая поисковая система MEDIE[9] [6], основанная на совмещении техник, связанных с анализом текстов и обработкой естественных языков для поиска в базе данных MEDLINE. Данная система предоставляет ряд новых возможностей:

1. Выполняет семантический поиск [7];
2. Выполняет поиск, основанный на общем списке согласований [8], позволяющий построить запрос с указанием конкретного поля в XML файле;
3. Извлекает описания генетических структур и названия заболевания.

На рис. 3 приведен результат семантического поиска по запросу “cause cancer” с извлечением названий заболеваний и описаний генетических структур.

[9] http://www.nactem.ac.uk/tsujii/medie/

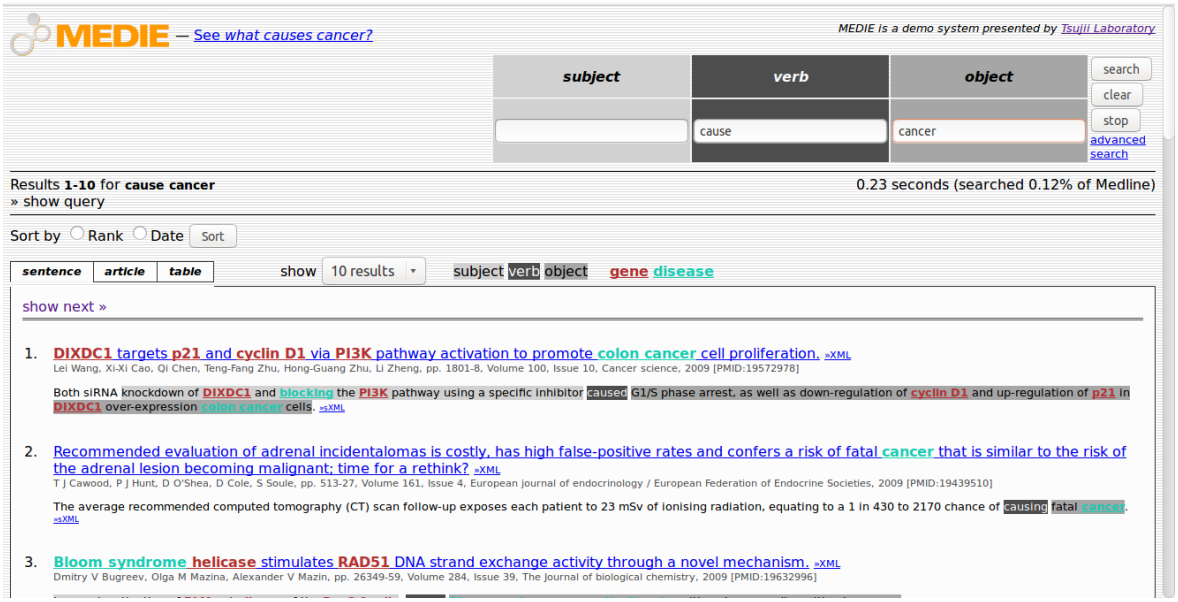

Рис. 3. Результаты семантического поиска в системе MEDIE.

Появившиеся возможности семантического поиска и извлечения фактов из медицинских исследований позволяют по-новому выполнять поиск к базе данных MEDLINE. Однако, по-прежнему результаты поиска системы MEDIE ранжируются без учета уровня доказательности найденного исследования.

Таким образом, основываясь на том, что задача ранжирования медицинских исследований по уровням доказательности на данный момент является открытой и востребованной, было принято решение начать разработку на базе Санкт-Петербургского государственного университета и при поддержке Первого Санкт-Петербургского государственного медицинского университета имени академика И. П. Павлова и кафедры офтальмологии Ташкентского института усовершенствования врачей поисковой системы для базы данных MEDLINE. Новизна и ценность данной системы заключается в методе ранжирования релевантных статей. Предлагается выполнять ранжирование релевантных статей, основанное на уровнях доказательности исследований с дополнительным учетом критериев оценки системы GRADE.

Постановка задачи

Объектом анализа данной работы являются документы, содержащие аннотации к статьям из международной базы данных медицинских исследований MEDLINE. Целью данного исследования является разработка алгоритма ранжирования медицинских исследований по уровням доказательности.

Поскольку одним из главных параметров ранжирования в данной задаче является уровень доказательности, то было решено начать разработку алгоритма ранжирования с классификацией медицинских исследований по уровням доказательности.

Необходимо заметить, что далее в работе будут рассматриваться 1, 2, 3 и 5 уровни доказательности поскольку на практике очень редко встречаются исследования 4 и 6 уровней, и обучение алгоритмов классификации и ранжирования по данным уровням становится затруднительным. Исходя из требований к РКИ, описанных во Введении, более детально представим 2 и 3 уровень доказательности следующим Списком 1:

1. Рандомизированные с двойным ослеплением
2. Рандомизированные с одинарным ослеплением
3. Рандомизированные открытые
4. Нерандомизированные с двойным ослеплением
5. Нерандомизрованные с одинарным ослеплением
6. Нерандомизированные открытые

Необходимо заметить, что во Введении мы одним из требований к РКИ указали применение техники ослепления к рандомизированной группе. В Списке 1 более детально представлены возможные комбинации РКИ с рандомизацией (с 1 по 3), соответствующие 2 уровню доказательности и РКИ без рандомизации (с 4 по 6), относящиеся к 3 уровню доказательности.

Для понимания комбинаций рандомизации и ослепления приведем более детальное описание техник ослепления. Техника двойного ослепления

подразумевает, что пациент/группа пациентов не знает о принимаемом медицинском вмешательстве, и медицинский эксперт не знает какое именно медицинское вмешательство принимает пациент/группа пациентов. В случае с одинарным ослеплением только пациент/группа пациентов не знает о принимаемом медицинском вмешательстве. Открытое ослепление не скрывает от пациента/группы пациентов информацию о принимаемом медицинском вмешательстве.

В качестве дополнительного параметра при ранжировании было решено учитывать подтип медицинского вмешательства. Для этого необходимо выполнять классифицирование медицинских исследований по подтипам медицинских вмешательств. Подтипами медицинских вмешательств являются различные методы для лечения и профилактики пациентов. Примеры подтипов медицинских вмешательств были взяты из данных ресурса Clinicaltrials.gov[10]. Данный сайт является государственным реестром, одобренным Международным комитетом редакторов медицинских журналов США, предоставляющим актуальную структурированную информацию о проведении клинических исследований широкого спектра заболеваний.

Далее будет учитываться следующий список подтипов медицинских вмешательств Список 2:

1. Лекарственные препараты;
2. Устройства;
3. Биологические препараты;
4. Процедуры;
5. Радиация;
6. Поведенческие (психотерапия);
7. Генетические;
8. Пищевые добавки.

[10] https://clinicaltrials.gov/

Учет данного параметра при ранжировании необходим, поскольку он позволяет локализовать поисковые результаты относительно определенного подтипа медицинского вмешательства, указанного в запросе. Например, в случае запроса “ingavirin influenca” на первом месте будут все исследования, которые принадлежат к подтипу медицинского вмешательства “Лекарственные препараты”.

Необходимо заметить, что результаты, полученные при классификации по подтипам медицинских вмешательств, дополнительно будут использоваться для задачи извлечения фактов. Решение данной задачи позволит применить автоматическое аннотирование к результатам поисковых запросов для разрабатываемой поисковой системы.

Поставленные выше задачи классификации относятся к методам машинного обучения и требует реализации следующих вспомогательных подзадач:

1. Разработка метода автоматической разметки обучающего множества аннотаций по уровням доказательности и подтипам медицинских вмешательств, основанного на существовании связи между документами, являющимися аннотациями к статьям базы MEDLINE, и содержимым зарегистрированных исследований в clinicaltrials.gov. Связь представлена расположенной в документе ссылкой на ресурс clinicaltrials.gov.
2. Решение проблемы, связанной с несбалансированным обучающим множеством, применяя методы, основанные на статистике (SMOTE [9]), и генеративные вероятностные модели (LDA [10]).
3. Обучение линейных методов мульти-классификации из выбранного набора классических алгоритмов Multinomial Naive Bayes; Multinomial Logistic Regression; Linear SVM из библиотеки sklearn[11]. Обучение ансамблей классификаторов Random Forest, Gradient Boosting Machine, AdaBoost on Random Forest и нелинейных алгоритмов классификации SVM

[11] http://scikit-learn.org

с полиномиальными ядрами и с RBF ядрами из библиотеки weka[12] для дальнейшего проведения оценки и выбора более эффективного метода.

4. Построение языковых *n*-граммных моделей и обучение частного случая марковских случайных полей – модель CRF [11], для извлечения фактов о подтипах медицинских вмешательств.
5. Формирование поискового индекса и разработка распределенной поисковой системы.

[12] http://www.cs.waikato.ac.nz/~ml/weka/

Обзор литературы

Несмотря на то, что на данный момент существует ряд поисковых систем, которые решают широкий спектр пользовательских запросов к базе данных MEDLINE, задача ранжирования медицинских исследований с точки зрения доказательной медицины до сих пор не является окончательно решенной. Однако, существует целый ряд исследований, в которых были проведены эксперименты с разработкой алгоритмов ранжирования и поиска надежных и научно строгих медицинских исследований с точки зрения доказательной медицины. Например, в работе [12] комбинируются оценки релевантности, полученные по модели Окапи BM25 [13], а также оценка качества, полученная с помощью Linear Support Vector Machines (SVM) для разработки алгоритма ранжирования медицинских исследований по уровням доказательности. В работе [12] вычисления показателя качества ранжирования рассчитывается как результат классификации аннотаций к статьям базы данных MEDLINE по двум классам: «Первоначальное исследование или обзорная статья, описывающая лечение, выполненное в строго научном стиле» и «другие». Более подробная информацию о проблеме классификации аннотаций MEDLINE по двум классам («научно строгий» и «научный, не строгий») представлены в работе [14]. В указанных экспериментах исследования проводятся с набором следующих алгоритмов классификации: Naive Bayes, SVM с полиномиальным ядром, boosting с деревом решений и stacking. Также существуют исследования, применяющие дополнительные структуры данных для ранжирования. Так, в работе [15] описывается эксперимент, изменяющий стратегию ранжирования, используя при этом структуру данных «термин-граф», оценивающую важность документа для запроса пользователя к базе данных MEDLINE. В исследовании [16] описывается инструмент, позволяющий генерировать систематические обзоры [17]. Одним из компонентов этого инструмента является классификатор MEDLINE статей по следующим типам исследований:

рандомизированное контролируемое исследование, обзорная статья, доказательное исследование, клинические рекомендации и другие [18]. Типы исследования дают более общее описание уровней доказательности. В то же время существует ряд исследований [19, 20], в которых MEDLINE статьи классифицируются по двум классам – рандомизированным и нерандомизированным контролируемым исследованиям.

Стоит отметить работы, в которых рассматриваются и разрабатываются алгоритмы машинного обучения для анализа базы данных MEDLINE, основанные на обучении без учителя. Существуют исследования, сравнивающие алгоритмы кластеризации, позволяющие сформировать кластеры из статей базы данных MEDLINE по различным заболеваниям [21]. В проделанной нами работе [22] сравнивались стандартные алгоритмы кластеризации: K-means, sequential information bottleneck, Hierarchical clustering, K-means ++ совместно с алгоритмами выбора признаков: LSA, Mutual Information для кластеризации MEDLINE аннотаций по подтипам медицинских вмешательств. В некоторых исследованиях приводятся описания новых алгоритмов кластеризации и проводятся эксперименты со статьями из базы данных MEDLINE. Так например, в работе [23] описывается новый подход к кластеризации – алгоритм SOPHIA, основанный на использовании аттракторов в качестве центроидов кластеров. В исследовании [24] приводится новый алгоритм COBRA, основанный на представлении документа как двудольного графа, где в одной доли расположены документы, а в другой – концепты, описывающие документы. Так же широко применяется кластеризация медицинских статей MEDLINE для задач доказательной медицины. Например, в исследовании [25] комбинируются техники аннотирования и информационного поиска для решения задачи поиска наилучшего лекарства для лечения заболевания, основываясь на уровнях доказательности.

Также хочется рассмотреть основные способы представления документов, описывающих медицинские исследования. Из-за сложности и

неоднозначности медицинской терминологии, все медицинские исследования описываются концептами применяя такие ресурсы, как MeSH[13] и UMLS Metathesaurus[14]. Подход, позволяющий описывать медицинские исследования в виде концептов, на данный момент применяется в поисковой системе PubMed для поиска релевантных медицинских статей. В работах [26, 27] сравниваются оценки релевантности результатов поиска для базы данных MEDLINE при использовании представлений медицинских статей “мешком концептов” (PubMed) и “мешком слов” (GoogleScholar[15]). В исследованиях [28, 29] сравниваются результаты ранжирования оценок релевантности, полученных при использовании комбинации представления медицинских статей “мешком слов” и “мешком концептов”. В работе [30] предложен фреймворк, позволяющий для каждого нового запроса обучать веса оценок релевантности, полученные при помощи представления медицинских статей “мешком слов” и “мешком концептов”.

В области обработки естественных языков также проводятся исследования, связанные с доказательной медициной и базой данных MEDLINE. В работе [31] проводятся эксперименты с извлечением фактов с позиции доказательной медицины из аннотаций к статьям базы данных MEDLINE, описывающих III фазу клинических исследований. В работе [32] рассматривается подход, позволяющий выполнять оценку автоматического аннотирования медицинских статей из базы данных MEDLINE с позиции доказательной медицины. В статье [33] рассматриваются иерархические скрытые условные случайные поля для извлечения информации из аннотаций базы данных MEDLINE, посвященных описанию генетических структур. В исследовании [34] рассматривается автоматический метод извлечения

[13] https://www.nlm.nih.gov/mesh/

[14] https://www.nlm.nih.gov/pubs/factsheets/umlsmeta.html

[15] https://scholar.google.ru/

терминов, специально разработанный для индексации документов в больших медицинских коллекциях, эксперименты проводятся на MEDLINE статьях.

Глава 1. Обзор алгоритмов

1.1. Описание применяемых алгоритмов

Поскольку в данной работе выполнялась классификация текстов, то выбирались алгоритмы, способные быстро обучаться на больших объемах данных, представляемых в *n*-мерных пространствах (*n* – число признаков). В связи с этим были выбраны линейные алгоритмы машинного обучения, позволяющие обучаться за линейное время с линейной памятью:

Support Vector Machines (SVM) – данный алгоритм был предложен В. Вапником и А. Червоненкисом в 1963 году [35]. Идея алгоритма заключается в построении оптимальной разделяющей гиперплоскости между двумя классами в n-мерном пространстве. Хотя данный алгоритм разработан для задачи бинарной классификации, необходимо заметить существование таких методов как: One-Vs-One и One-vs-Rest [36], позволяющих применить данный алгоритм к решению задачи много-классовой классификации. В данной работе дополнительно применялся модифицированный нелинейный SVM, основанный на поиске нелинейной (ядерной) разделяющей поверхности. В качестве ядер использовались полиномиальные и RBF ядра. Идея нелинейной классификации SVM была предложена Б. Босером, И. Гуйон и В. Вапником в 1992 году [37]. Также в работе применялась модификация One-class-SVM для одно-классовой классификации [38]. Данная модификация SVM активно применяется для детектирования выбросов в данных.

Naive Bayes classifier – является популярным алгоритмом для классификации и категоризации текстов, появившимся в начале 1960-х годов [39]. Основная идея алгоритма основана на теореме Байеса для нахождении условных вероятностей. Задача данного классификатора заключается в восстановлении *n* одномерных плотностей для каждого класса. Данный алгоритм не требует применения дополнительных методов для решения задачи многоклассовой классификации.

Logistic regression classifier – данный алгоритм является частным, робастным случаем линейной регрессии [40]. Основная идея заключается в модификации линейной регрессии применением logit (сигмоидной) преобразовывающей функции, позволяющей получать предсказания на интервале [0,1]. Данный алгоритм хорошо масштабируется для решения задачи много-классовой классификации.

В ходе проведения экспериментов решались проблемы обучения на несбалансированных данных применялись мета и ансамблевые методы классификации. Решение было основано на том, что на практике [41, 42] при работе с несбалансированными данными приведенные ниже алгоритмы показывали лучшие результаты.

Идея алгоритма **Random Forest** была предложена Т.К. Хо в 1995 году [43], сам алгоритм был разработан в 1996 году Л. Брейманом [44]. Основная идея алгоритма заключается в построении деревьев решений на случайных подвыборках, взятых из обучаемого множества. Выбор класса для объекта происходит по итогам голосования построенных деревьев. Данный алгоритм активно применяется на практике и с увеличением числа деревьев в лесу не приводит к переобучению, а лишь улучшает качество классификации.

Алгоритм **Adaptive Boosting (AdaBoost)** был предложен Й. Фройндом и Р. Шапиром в 1997 году [45]. AdaBoost является мета-алгоритмом, основная идея которого заключается в итерационном применении слабых алгоритмов классификации. На каждой итерации в цикле новый алгоритм старается классифицировать объекты, с которыми не справился предыдущий. В данной работе применялся AdaBoost с набором из Random Forest.

Для оценки подбора гиперпараметров алгоритмов классификации применялись следующие подходы.

Random Layout – подход для подбора гиперпараметров для алгоритмов машинного обучения [46]. Данный подход является итерационным, и его основная идея заключается в том, что на новом шаге каждый гиперпараметр задается распределением, из которого он выбирается случайным образом.

Далее применяется метод скользящего среднего с *n*-блоками для проверки качества данного параметра на обучающем множестве.

Скользящее среднее с *n*-блоками – метод машинного обучения, оценивающий качество алгоритмов классификации. Данный метод позволяет получить оценки без переобучения на тренировочном множестве. Основная идея заключается в том, что мы разбиваем обучающее множество на *n* блоков. Далее на *n*-1 блоках мы обучаемся, а на *n*-ом – тестируем обученную модель. Такую последовательность действий мы выполняем *n* раз, после чего усредняем оценки и получаем оценку качества алгоритма без переобучения.

Дополнительно в данной работе для решения проблемы несбалансированных данных применялись методы, основанные на генерации повторных выборок (resampling) и моделировании топиков.

Synthetic Minority Over-sampling Technique (SMOTE) – статистический метод, активно применяемый на практике для решения проблемы не сбалансированных данных [9]. Данный подход основан на применении генерации повторных выборок (resampling) выборки совместно с методом *n* ближайших соседей. Выбирая случайным образом объект из выборки, мы дополнительно применяем метод ближайших соседей для выбора дополнительных объектов и, таким образом, через заданное число итераций формируем новую выборку с необходимым для нас объемом. SMOTE эффективно применяется для понижения объема класса с большим числом объектов. Однако в случае с классами меньших объемов применение SMOTE для увеличения объема класса приводит к увеличению дубликатов и переобучению.

Latent Dirichlet allocation (LDA) – генеративная вероятностная модель, предложенная в 2003 году Д.М. Блеием, А.И. Нджи и М.И. Жорданом [10]. Данная модель позволяет представлять документы смесью топиков, описывающих документ. Основная идея LDA заключается в предположении, что распределение топиков может быть представлено распределением Дирихле. Данная модель активно применяется на практике для задач

аннотирования, классификации и кластеризации в качестве нового способа представления документов.

Для задачи понижения размерности пространства признаков применялся следующий алгоритм:

Latent semantic analysis (LSA) – данный алгоритм был предложен Ж.П. Бенжекри в 1970 году [47]. Основная идея алгоритма заключается в применении SVD разложения матрицы термин-документ.

Для решения задачи извлечения фактов применялись следующие подходы:

Conditional Random Fields (CRF) – является графическим вероятностным алгоритмом, частным случаем Марковских случайных полей [11]. Основная идея заключается в определении условного распределения вероятностей по последовательности тегов, учитывая заданную последовательность наблюдений. На практике данный подход активно применяется в области обработки естественных языков для извлечения фактов и создания грамматической разметки.

***N*-граммная модель** – языковая модель, применяемая в области обработки естественных языков для предсказания слова с учетом условной вероятности *n* предыдущих [48]. Основная идея построения данной модели заключается в подсчете условных вероятностей для всех возможных *n*-грамм в корпусе текстов. В случае обучения *n*-граммной модели на большом объеме текстов модель может применяться для генерации текстов. Частным случаем применения *n*-граммной модели является решение задачи извлечения фактов из текста.

1.2. Используемые метрики качества

Для оценки качества работы классификаторов применялись следующие метрики: *точность*, *полнота* и *F-мера*.

$$\text{Точность} = \frac{\text{ИП}}{\text{ИП} + \text{ЛП}}$$

$$\text{Полнота} = \frac{\text{ИП}}{\text{ИП} + \text{ЛО}}$$

$$F - \text{мера} = 2 * \frac{\text{Точность} * \text{Полнота}}{\text{Точность} + \text{Полнота}}$$

где *ИП* – истинно положительные значения классификации, классификатор правильно соотнес элемент тестовой выборки. *ЛП* – ложно положительные, классификатор ошибочно соотнес элемент к классу. *ЛО* – ложно отрицательные элементы, классификатор ошибочно не соотнес элемент к классу.

Для оценки мульти-классификации алгоритмов, высчитывались *макро-* и *микроточность*, *полнота* и *F-мера* по следующим формулам.

Макро:

$$\text{Макро точность} = \frac{\sum_{n=1}^{m} \text{Точность}_n}{m}$$

$$\text{Макро полнота} = \frac{\sum_{n=1}^{m} \text{Полнота}_n}{m}$$

$$\text{Макро } F - \text{мера} = 2 * \frac{\text{Макро Точность} * \text{Макро Полнота}}{\text{Макро Точность} + \text{Макро Полнота}}$$

Микро:

$$\text{Микро точность} = \frac{\sum_{n=1}^{m} \text{ИП}_n}{\sum_{n=1}^{m} \text{ИП} + \sum_{n=1}^{m} \text{ЛП}}$$

$$\text{Микро полнота} = \frac{\sum_{n=1}^{m} \text{ИП}_n}{\sum_{n=1}^{m} \text{ИП} + \sum_{n=1}^{m} \text{ЛО}}$$

$$\text{Микро } F - \text{мера} = 2 * \frac{\text{Микро Точность} * \text{Микро Полнота}}{\text{Микро Точность} + \text{Микро Полнота}}$$

где m – количество классов, а вычисление микро полноты и точности выполняется суммированием всех истинно положительных, ложно положительных и ложно отрицательных результатов работы классификации для каждого класса.

1.3. Предварительная обработка аннотаций

Для применения алгоритмов классификации, описанных в главе 1, использовалось представление аннотаций моделью “мешок слов” [7]. Для этого производилась токенизация с удалением стоп слов и выполнением стемминга. Стемминг выполнялся с применением алгоритма Портера.

В качестве вектора признаков для аннотаций использовался вектор слов из словаря, составленного по всему корпусу аннотаций. Вес каждого слова в аннотации оценивался метрикой tf-idf, где tf (term frequency) – частота термина, idf (inverse document frequency) - инвертированная частота документа.

$$tf(t,d) = \frac{n_i}{\sum_k n_k}$$

где t – слово, d – документ, в числителе располагается число вхождений слова в документ, а в знаменателе — общее число слов в данном документе.

$$idf(t,D) = log\frac{|D|}{|(d_i \supset t_i)|}$$

где, |D| — количество документов в корпусе; $|d_i \supset t_i|$— количество документов, в которых встречается слово t_i.

1.4. Архитектура поисковой системы

Поскольку целью является разработка поисковой системы по базе данных MEDLINE, ранжирующей поисковые результаты по уровням доказательности, то необходимо привести архитектуру разрабатываемой поисковой системы (рис. 4).

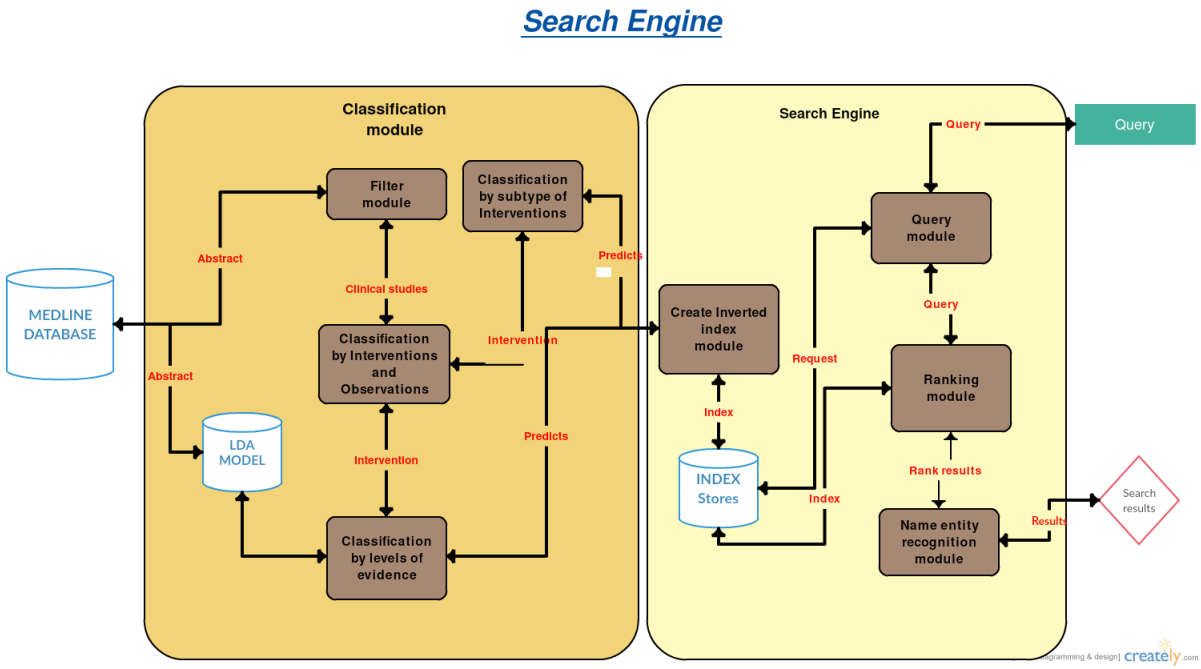

Рис. 4. Архитектура поисковой системы.

Первым модулем системы является Модуль классификации, позволяющий определить подтип медицинского вмешательства и уровень доказательности исследования. Модуль классификации состоит из четырех основных частей:

1. Фильтр – позволяющий отсеивать выбрать только клинические исследования для дальнейшей классификации;
2. Классификатор аннотаций по исследованиям «с вмешательствами» и «обзорным» – разделяет обзорные аннотации от аннотаций с вмешательством для дальнейшей передачи аннотаций с вмешательством в Классификатор по подтипам медицинских вмешательств и Классификатор по уровням доказательности;
3. Классификатор по уровням доказательности;
4. Классификатор по подтипам медицинских вмешательств;
5. Обученная модель LDA.

Более подробное описание модуля классификации представлено в Главе 3.

Вторым основным модулем является Поисковой модуль, состоящий из следующих частей:

6. Модуль построения инвертированного индекса [7];
7. Хранилище инвертированного индекса;
8. Модуль обработки запросов – выполняет нормализацию запроса и обращается к хранилищу инвертированного индекса для получения аннотаций по запросу. Полученные аннотации передает в модуль ранжирования;
9. Модуль ранжирования – ранжирует результаты, предоставленные модулем обработки запросов, учитывая уровень доказательности и подтип медицинского вмешательства;
10. Модуль извлечения именованных сущностей – извлекает факты из аннотаций, например: “описание подтипа медицинского вмешательства”, “заболевания”.

Более детальное описание поискового модуля содержится в Главе 4.

Глава 2. Подготовка обучающего множества

2.1. Подготовка обучающего множества для задачи классификации

Для решения задачи разметки корпуса аннотаций на первом этапе рассматривались 90 документов 2011 года, являющихся аннотациями к статьям базы данных MEDLINE. Для удобной обработки аннотаций была создана следующая структура:

```
<document>
    <doc_id></doc_id>
    <date></date>
    <title></title>
    <body></body>
    <topics></topics>
    <place></place>
    <author></author>
          <type></type>
<evidence></evidence>
<intervention_name>
<intervention_type>
</document>
```

Структура документа (1)

где: <document> – контейнер документа; <doc_id> – идентификатор статьи; <date> – дата публикации статьи; <title> – заголовок статьи; <body> – тело аннотации статьи; <topics> – ключевые слова в статье; <place> – журнал, опубликовавший статью; <author> – авторы статьи; <type> – подтип медицинского вмешательства; <evidence> – уровень доказательности; <intervention_name> – наименование медицинского

вмешательства; <intervention_type> – вмешательства (обзорные/ с медицинским вмешательством).

Данные 90 документов были размечены вручную, основываясь на поиске связи между аннотациями и ресурсом clinicaltrials.gov. Связь представляла собой нахождение в аннотации ссылки формата NCT000000000 (например, NCT00893711), означающей индексацию данного исследования на clinicaltrials.gov. Последовательность ручной разметки была следующей:

1. Осуществлялся поиск ссылки в аннотации на clinicaltrials.gov;
2. При помощи внутренней поисковой системы сайта clinicaltrials.gov производился поиск исследования по найденной из аннотации ссылке (Рис. 5).
3. Найденные в результате поиска уровень доказательности и подтип медицинского вмешательства вручную добавлялись в структуру документа (1).

На результаты поиска в ресурсе clinicaltrials.gov было предложено наложить ограничения:

1. В случае если исследование описывает тип рандомизации (кластерная, блочная, стратифицированная и т.д.), оставлять только значение с рандомизацией/без рандомизации без указания типа;
2. В случае если исследование представлено двумя подтипами медицинского вмешательства (Рис. 5) помечать аннотацию первым подтипом, поскольку он несет в себе главную информацию об исследовании.

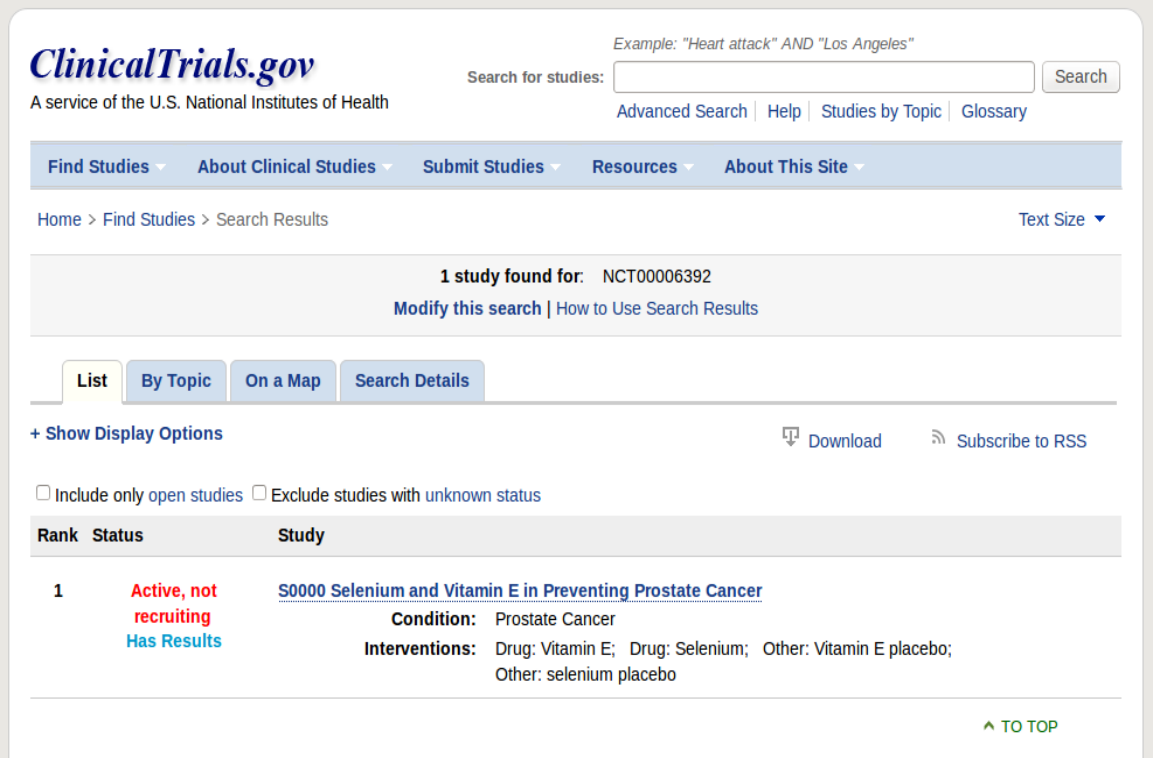

Рис.5. Результат поиска по ссылке на сайте clinicaltrials.gov.

В результате ручной разметки из 90 аннотаций удалось разметить 60. Оставшиеся 30 поделились на группы:

1. не содержащие ссылку на clinicaltrials.gov;
2. содержащие ссылку, но не имеющие информации об уровне доказательности и о подтипе медицинского вмешательства;
3. содержащие ссылку с ошибкой (I00CTN51481987).

Далее было решено рассматривать корпус из 2000000 аннотаций с 2006 по 2013 года и автоматизировать процесс разметки. Автоматизация была реализована при помощи разработанного скрипта на Python. Python скрипт выполняет следующие действия:

1. поиск в аннотациях ссылки типа NCTXXXXXXXX как регулярного выражения;
2. web-crawling, осуществляющий перебор ссылок http://clinicaltrials.gov/show/NCTXXXXXXXX?resultsxml=true, подменяя часть (NCTXXXXXXXX) на найденную в аннотации;

3. извлекаючение из xml страницы данные, содержащиеся в полях <study_design> и <intervention_type> дополнительно извлекалось поле <intervention_name> для обучения методов извлечения фактов. На результат разбора xml страниц также накладывались ограничения: в случае обнаружения двух полей <intervention_type> аннотация помечалась первым.

В результате автоматической разметки из 2000000 аннотаций удалось разметить 6123 по уровням доказательности и 3534 по подтипам медицинских вмешательств. Оставшиеся не размеченные аннотации:

1 - не содержали ссылку на clinicaltrials.gov;

2- содержали ссылку, но при переходе web-crawler на соответствующую страницу появлялась ошибка “404 - страница не найдена”;

3- содержали ссылку с ошибкой (например, CTNO1481987).

В результате каждый уровень доказательности содержал следующее количество аннотаций, представленное в табл. 3.

Таблица 3. Результаты разметки по уровням доказательности

Класс	**Число аннотаций**
Рандомизированные с одинарным ослеплением	883
Рандомизированные с двойным ослеплением	2740
Рандомизированные открытые	1955
Нерандомизированные с одинарным ослеплением	20
Нерандомизированные с двойным ослеплением	12
Нерандомизированные открытые	513

Результаты разметки по подтипам медицинских вмешательств представлены в табл. 4.

Таблица 4. Результаты разметки по подтипам медицинских вмешательств

Класс	Число аннотаций
Лекарственные препараты	1619
Устройства	238
Биологические препараты	242
Процедуры	300
Радиация	18
Поведенческие (Психотерапия)	585
Генетические	8
Пищевые добавки	191
Прочие	333

Дополнительно удалось сформировать выборку из размеченных аннотаций по двум уровням: в классе “Обзорные исследования” - 741 аннотация и “Исследования с медицинским вмешательством” - 7817 аннотаций. На данной странице[16] приведена ссылка на Python скрипт, размещенный на GitHub, выполняющий процесс автоматической разметки аннотаций.

2.2. Подготовка обучающего множества для задачи извлечения фактов

Для задачи извлечения фактов отдельно подготавливалась выборка для извлечения информации о наименовании лекарственного препарата, применяемого в медицинском исследовании. На данном этапе рассматривались только аннотации, размеченные типом медицинского

[16] https://github.com/KamalovMikhail/ResearchNLP/blob/master/SearchNCT.py

вмешательства Лекарственный препарат (класс Лекарственный препарат содержит 1619 аннотаций), полученные на предыдущем этапе автоматической разметки. Далее выполнялась дополнительная обработка данных аннотаций:

1. Разделение аннотаций на предложения и последующий их синтаксический анализ. Получение лексико-грамматических меток для каждого слова, используя Stanford NLP tagger[17]. Пример лексико-грамматической разметки представлен на Рис. 6.
2. Поиск по ключевым словам названий “Лекарственного препарата” и их разметка drug (1) / not drug (0).

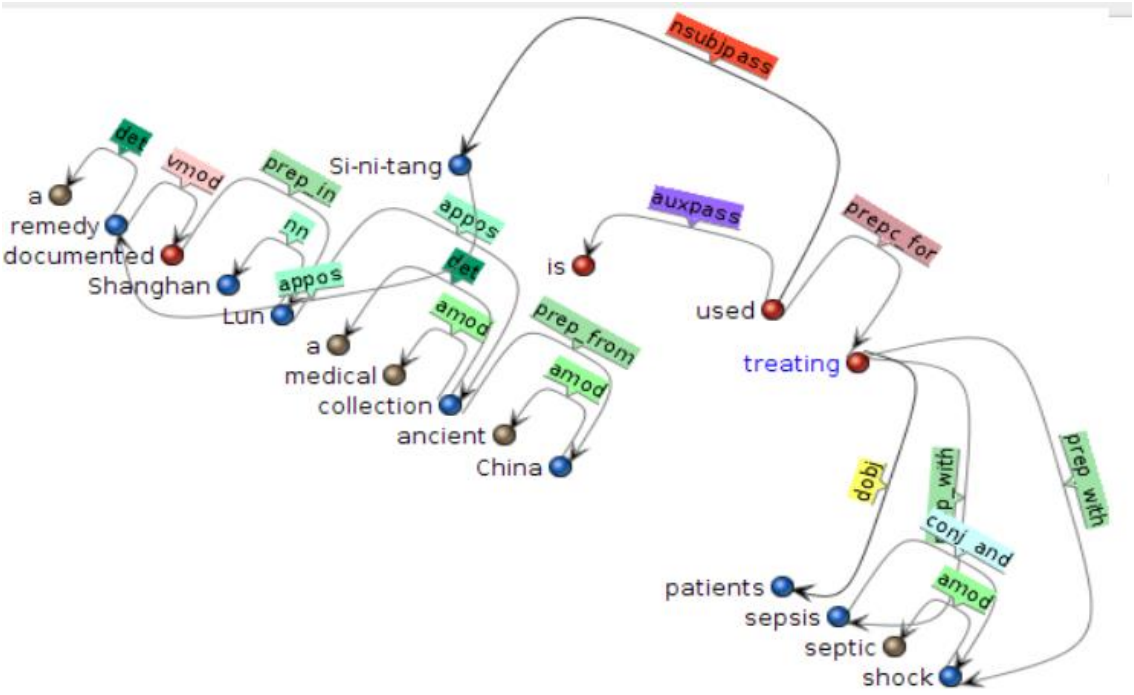

Рис. 6. Пример лексико-грамматической разметки.

Поскольку в данной работе мы применяли лексико-грамматическую разметку, то сохранялись стоп слова и начальные формы слова без применения стемминга. В результате обучающая выборка содержала 609825 слов, из них 13098 описывало название лекарственных препаратов. Ссылки на python

[17] http://nlp.stanford.edu/software/tagger.shtml

скрипты[18] подготовки обучающего множества и описание[19] его применения выложены на ресурсах GitHub и Bitbucket.

2.3. Генерация синтетических аннотаций при помощи модели Latent Dirichlet Allocation

В результате формирования обучающих множеств можно заметить несбалансированное число объектов в классах. Для этой задачи было решено применить модель LDA для генерации синтетических аннотаций. Основная идея заключалась в том, что можно аннотации из классов с меньшим объемом представлять топиками для дальнейшего выбора из этих топиков списка слов, которые с наибольшей вероятностью принадлежат к данному топику.

На начальном этапе было решено обучить модель LDA со 100 топиками на случайно выбранных 500000 аннотациях из 2000000 аннотаций с 2006 по 2013 года. Алгоритм LDA был взят из библиотеки gensim[20]. В табл. 5 приведены данные о скорости обучения алгоритма и параметрах компьютера.

Таблица 5. Результаты обучения модели LDA

Параметры компьютера	**Параметры модели LDA**	**Время**	**Число аннотаций**
Процессор: Intel® Core™ i5-4200U processor Dual-core 1.60 GHz **Оперативная память:** 8 GB, DDR3L SDRAM	**Чило потоков:** 4 **Число топиков**: 100 **Алгоритм обучения:** Variational inference	20 часов 40 минут	500000

[18] https://bitbucket.org/leadenhoar/medline/src/59d8dfa236ada4750a14107fa00285d2435ac746/learning_set/src/stanford_parser/?at=master

[19] https://gist.github.com/KamalovMikhail/d98bbec36d9363f83fbf1f3270cdaa17

[20] http://radimrehurek.com/

Следующим этапом стал процесс генерации синтетических аннотаций, состоящий из последовательного выполнения шагов:

1. Случайным образом выбираем аннотацию из класса с меньшим объемом.
2. Представляем аннотацию набором топиков.
3. Начинаем итерационный обход топиков, полученных на шаге 2.

3.1 Для каждого топика выбираем список из 100 слов с наибольшей вероятностью принадлежности слова топику.

3.2 Из полученного списка 100 слов случайным образом выбираем 10 слов и добавляем в синтетическую аннотацию.

4. В результате итерационного процесса получаем синтетическую аннотацию.

Необходимо заметить, что применение шага 3.2 позволяет избежать появления дубликатов последовательности слов в синтетических аннотациях.

Глава 3. Описание модуля классификации

3.1. Классификация MEDLINE аннотаций по уровням доказательности

В результате полученной разметки MEDLINE аннотаций по уровням доказательности (процесс формирования обучающего множества описан в пункте 1.1), было решено приступить к решению задачи классификации. В данной работе учитываются 1, 2, 3 и 5 уровни доказательности, при этом следует отметить, что автоматическая разметка выполнялась только для 2, 3 и 5 уровней. Автоматическая разметка не учитывала 1 уровень, поскольку все MEDLINE аннотации, содержащие в заголовке термин “мета-анализ”, было решено помечать 1 уровнем доказательности. Данное решение обусловлено тем, что проведение мета-анализа является трудоемким процессом, и авторы медицинского исследования всегда указывают в названии данный термин.

3.2. Фильтрация клинических исследований

Первым этапом разработки модуля классификации стало построение фильтра, отсеивающего аннотации, не являющиеся описанием клинических исследований. Поскольку уровни доказательности относятся лишь к описаниям клинических исследований. Под клиническим исследованием понимается исследование, в рамках которого проводится лечение/наблюдение за пациентом/группой пациентов. В связи с тем, что в выборках, сформированных в пункте 1.1, аннотации описывают только клинические исследования без примеров не клинических исследований, решение данной проблемы алгоритмами классификации стало невозможным. Исходя из этого, было решено построить фильтр на коллокациях (устойчивых словосочетаниях) для отсеивания аннотаций, описывающих не клинические исследования. Для выбора коллокаций нам необходимо было сформировать

список исследований, не являющихся клиническими. Для этого был выполнен ряд действий:

1. Обучить метод одно-классовой классификации One-class-SVM с RBF ядром из библиотеки sklearn[21] на имеющейся размеченной выборке из 6123 аннотаций по уровням доказательности.
2. Случайным образом выбрать 1000 аннотаций из 2000000 и получить для них предсказания на обученной модели One-class-SVM;
3. Основываясь на полученных на 2 шаге предсказаниях выбрать аннотации, которые были классифицированы как “не клинические исследования” и отправить данный результат на проверку медицинским экспертам;
4. Извлечь из одобренных медицинскими экспертами аннотаций коллокации при помощи библиотеки nltk[22].

Пример работы фильтра приведен на рис. 7.

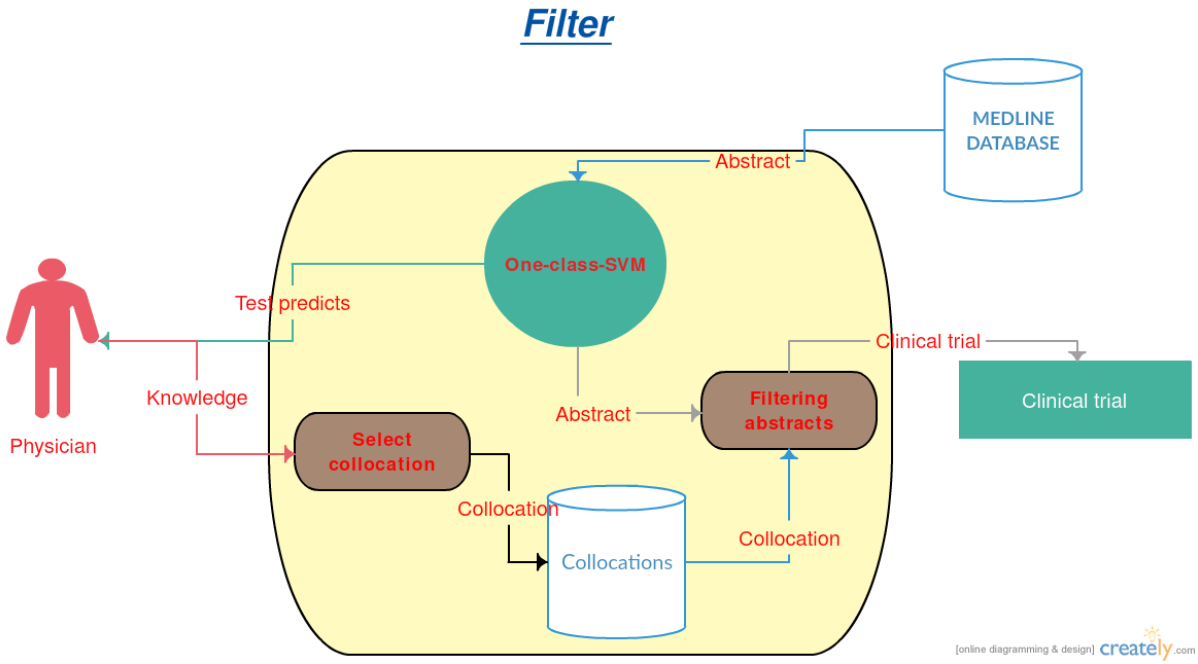

Рис. 7. Рабочая модель фильтрации MEDLINE аннотаций.

Стоит заметить, что идея One class SVM с RBF ядром основана на построении нелинейной гиперплоскости вокруг одного класса. Данный

[21] http://scikit-learn.org

[22] www.**nltk**.org/

подход активно применяется для детектирования выбросов в данных. В нашем случае не клинические исследования рассматриваются как выбросы.

В результате выполнения 2 шага из 1000 случайно отобранных аннотаций 200 были предсказаны как не описывающие клиническое исследование. Данные 200 статей были отправлены экспертам на оценку, из них 108 были подтверждены как исследования, не относящиеся к клиническим. Над проверенными 108 аннотациями выполнялась предобработка: удалялись стоп слова, применялся стемминг и выполнялось извлечение биграммных и триграммных коллокаций при помощи библиотеки nltk. Топ 5 коллокаций приведен в табл. 6.

Таблица 6. Извлеченные коллокации

Биграммные коллокации	**Триграммные коллокации**
activ normal	activ normal growth
approxim higher	bacterium investig effect
bacterium investig	contrast activ normal
contrast activ	cultiv inorgan solut
cultiv inorgan	fischeri harveyi character

Полученные коллокации применялись для фильтрации результатов предсказания One class SVM классификатора. Фильтрация заключалась в проверке аннотаций, классифицированных как клинические исследования. Таким образом, если в фильтр попадала аннотация, классифицированная как клиническое исследование, и при этом в ней была найдена коллокация, описывающая не клиническое исследование, то данная аннотация не проходила через фильтр.

На данный момент построенный фильтр применяется для отсеивания не клинических исследований без этапа тестирования медицинскими экспертами как часть Модуля классификации.

3.3. Классификации аннотаций на “обзорные” и “с вмешательством”

Поскольку второй уровень доказательности принадлежит исследованиям, которые удовлетворяют всем требованиям проведения РКИ, а третий – исследованиям с частичным выполнением требований РКИ (без рандомизации), то можно заметить, что 2 и 3 уровень доказательности принадлежит исследованиям, в которых выполняется медицинское вмешательство, в то время как 5 уровень занимают обзорные исследования. Исходя из этого полученные в пункте 2.1 клинические исследования было решено подобрать классификатор для разделения аннотаций на обзорные и с медицинским вмешательством. Поскольку в главе 1 была получена выборка размеченных аннотаций на Обзорные исследования – 741 аннотаций и Исследования с медицинским вмешательством – 7817 аннотаций то было принято решение провести эксперименты с бинарными алгоритмами классификации. В качестве бинарных классификаторов был выбран ряд линейных алгоритмов: Linear SVM, Logistic regression classifier, Naive Bayes classifier. Выбор линейных классификаторов был основан на линейной скорости обучения и требуемой линейной памяти.

В главе 4 (пункт 4.1) подробно описаны результаты экспериментов классификации MEDLINE аннотаций на исследования “с вмешательством” и “обзорные” с комбинацией различных методов балансировки обучающего множества.

В результате экспериментов было получено, что лучшее качество классификации показал Linear SVM при балансировке обучающего множества синтетическими аннотациями (табл. 7). Лучшие гиперпараметры Linear SVM:

penalty = ‘l2’, loss='squared_hingle', multi_class=’ovr’ , C = 1.0

Таблица. 7. Результаты Linear SVM

Класс	**Точность**	**Полнота**	**F-мера**
С вмешательством	0.9364	0.7505	0.8332
Обзорные	0.9291	0.7654	0.8394

Использование синтетических аннотаций для балансировки обучающего множества повышает качество классификаторов. Полученный результат защищен от недообучения тем, что мы увеличиваем объем класса с меньшим числом объектов до объема класса с большим числом.

3.4. Классификации аннотаций по 2 и 3 уровням доказательности

Основываясь на требованиях к РКИ, описанным во Введении, для более детального представления 2 и 3 уровней доказательности воспользуемся Списком 1:

1. Рандомизированные с двойным ослеплением;
2. Рандомизированные с одинарным ослеплением;
3. Рандомизированные открытые;
4. Нерандомизированные с двойным ослеплением;
5. Нерандомизрованные с одинарным ослеплением;
6. Нерандомизированные открытые.

Детальное описание уровней Списка 1 приведено в разделе Постановка задачи. Учитывая применение более точного описания 2 и 3 уровней доказательности в Модуль классификации было решено включить классификатор, выполняющий мульти-классификацию аннотаций по Списку 1. Основываясь на исследованиях [41, 42] для решения проблемы несбалансированных данных в мульти-классовой классификации были выбраны для сравнения два алгоритма: SVM на RBF ядрах и AdaBoost в сочетании с Random Forest.

Принимая во внимание тот факт, что обучающее множество (табл. 3), полученное в Главе 1 пункт 1.1, является не сбалансированным, и уровни в Списке 1 можно представить комбинацией двух классов (1 - Рандомизированные / Нерандомизированные; 2 - Двойное ослепление / Одинарное ослепление/ Открытое;), было решено выполнить декомпозицию уровней доказательности.

Декомпозиция уровней доказательности (рис. 8) позволяет разбить задачу классификации аннотаций по уровням доказательности (Список 1) на две независимые подзадачи:

- Классификация аннотаций по рандомизации;
- Классификация аннотаций по виду ослепления.

Решение данных подзадач позволяет избавиться от сильной корреляции между уровнями доказательности.

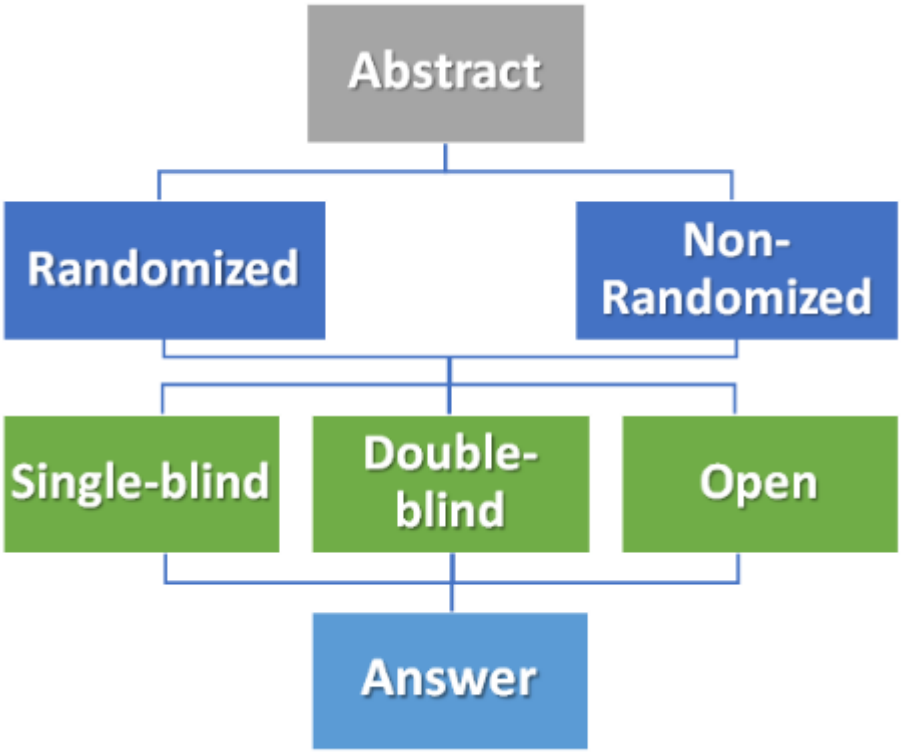

Рис. 8. Декомпозиция уровней доказательности.

Предложенный подход декомпозиции классов позволяет получить лучший результат классификации, выбирая технику балансировки обучающего множества и алгоритм классификации для каждой подзадачи в

отдельности с последующим слиянием результатов. В Главе 5 (пункт 5.2.) подробно описываются эксперименты по сравнению работы SVM на RBF ядрах и AdaBoost в сочетании с Random Forest для каждой подзадачи, дополнительно применяя подходы для балансировки обучающего множества. В экспериментах использовались реализации алгоритмов SVM на RBF ядрах и AdaBoost в сочетании с Random Forest из библиотеки WEKA[23].

В процессе проведения экспериментов лучший результат при балансировке обучающего множества синтетическими аннотациями для подзадачи 1 показал SVM на RBF ядрах с гиперпараметрами :

-S 0 -K 2 -D 3 -G 0.0 -R 0.0 -N 0.5 -M 40.0 -C 1.0 -E 0.001 -P 0.1.

Лучший результат для решения 2 подзадачи был также получен балансировкой обучающего множества синтетическими аннотациями и применением алгоритма AdaBoost в сочетании с RandomForest с гиперпараметрами:

-I 100 -K 0 -S 1.

В результате слияния результатов классификации были получены следующие оценки качества для уровней доказательности (табл. 8):

Таблица 8. Результаты классификации при декомпозиции уровней доказательности

Класс	**Точность**	**Полнота**	**F-мера**
Рандомизированные с двойным ослеплением	0.9264	0.7505	0.8232
Рандомизированные с одинарным ослеплением	0.8634	0.7211	0.7858
Рандомизированные открытые	0.9122	0.8264	0.8671
Нерандомизрованные с одинарным ослеплением	0.9234	0.7513	0.8335

[23] http://www.cs.waikato.ac.nz/~ml/weka/

Нерандомизированные с двойным ослеплением	0.8732	0.7531	0.8617
Нерандомизированные открытые	0.8935	0.7216	0.7984

Таким образом, декомпозиция уровней доказательности улучшает качество классификации аннотаций за счет того, что для каждой задачи по отдельности выполняется генерация синтетических аннотаций и подбирается лучший алгоритм классификации.

3.5. Классификация по подтипам медицинских вмешательств

Поскольку в качестве дополнительного параметра при ранжировании было решено учитывать подтип медицинского вмешательства, в Модуль классификации включался классификатор медицинских исследований по подтипам медицинских вмешательств. Детально описанный во Введении Список 2 представляет описание подтипов медицинских вмешательств. Список 2:

1. Лекарственные препараты;
2. Устройства;
3. Биологические препараты;
4. Процедуры;
5. Радиация;
6. Поведенческие (Психотерапия);
7. Генетические;
8. Пищевые добавки.

Для решения данной задачи были выбраны линейные алгоритмы классификации Linear SVM, Logistic regression classifier, Naive Bayes classifier. Данный список алгоритмов был выбран, исходя из результатов классификации аннотаций на “обзорные” и “с вмешательством”.

Поскольку в обучающем множестве, полученном в главе 1, содержатся классы с 12 аннотациями, и учитывая тот факт, что подтипы медицинских вмешательств нельзя разделить, как в случае с декомпозицией 2 и 3 уровней доказательности, применение методов для балансировки выборки стало не возможным. Однако применялся подход для выбора признаков LSA. Подробное описание результатов сравнения линейных алгоритмов классификации без применения методов балансировки обучающего множества для решения поставленной задачи приведено в главе 5.

В результате проведенных экспериментов (табл. 9) лучшее качество показал алгоритм Linear SVM без применения LSA с гиперпараметрами:

penalty = ‘l2’, loss='squared_hingle', multi_class=’ovr’ , C = 1.0

Таблица 9. Результаты классификации по подтипам медицинских вмешательств без LSA

Класс	**Точность**			**Полнота**			**F-мера**		
	Naive Bayes	Linear SVC	Logistic Regression	Naive Bayes	Linear SVC	Logistic Regression	Naive Bayes	Linear SVC	Logistic Regression
Поведенческие	0.7781	0.7366	0.7051	0.4956	0.7814	0.7612	0.60	0.7589	0.7316
Биологические препараты	1	0.89	0.96	0.2417	0.7245	0.65	0.3865	0.8	0.7715
Устройства	0	0.6274	0.7145	0	0.4152	0.1231	0	0.49	0.1867
Пищевые добавки	0	0.5812	0.58	0	0.6317	0.2466	0	0.60	0.3415
Лекарственные препараты	0.5312	0.7988	0.6512	0.9981	0.94	0.97	0.69	0.8628	0.7826
Генетические	1	0.34	0.50	1	0.2278	0.0917	0.0318	0.2758	0.1535
Процедуры	0	0.6479	0.65	0	0.42	0.1452	0	0.5188	0.2374
Радиация	0	1	0	0	0.3345	0	0	0.5067	0

В результате проведенных экспериментов можно заметить, что лучшую оценку качества показал алгоритм Linear SVM без применения LDA.

Глава 4. Описание поискового модуля

Для создания поискового модуля использовалась библиотека Lucene 4.2[24]. Lucene является библиотекой с открытым исходным кодом, предоставляющей большой список готовых решений для разработки поисковых систем.

4.1. Формирование инвертированного индекса

В библиотеке Lucene базовым объектом индексации является Field[25]. При помощи нескольких компонентов Field можно собрать объект Document[26]. Предоставляемая возможность создания объекта Document позволяет для разных объектов Field применять различную предобработку. Под предобработкой может пониматься например: токенизация (подход для выделения отдельных слов), удаление стоп слов, стемминг, лемматизация.

В модуле, формирующем инвертированный индекс, объектом индексации являлся Document (рис. 9), сформированный из следующего списка объектов Field:

1. Field: Title - содержит название аннотации. В качестве предобработки использовалась токенизация; в качестве разбиения применялись пробелы с удалением знаков препинания, удаление стоп слов и стемминг.
2. Field: Body - содержит текст аннотации. В качестве предобработки использовалась токенизация; в качестве разбиения применялись пробелы с удалением знаков препинания, удаление стоп слов и стемминг.
3. Field: MESH terms - содержит список MeSH терминов, описывающих аннотацию. В качестве предобработки использовалась токенизация; в качестве разбиения применялись пробелы с удалением знаков препинания.

[24] http://lucene.apache.org/

[25] https://lucene.apache.org/core/4_2_0/core/org/apache/lucene/document/Field.html

[26] https://lucene.apache.org/core/4_2_0/core/org/apache/lucene/document/Document.html

4. Field: Level of evidence - содержит информацию об уровне доказательности аннотации, полученном из модуля классификации. Предобработка не предполагается.
5. Field: Subtype of intervention - содержит информацию о подтипе медицинского вмешательства, полученную полученную из модуля классификации. Предобработка не предполагается.

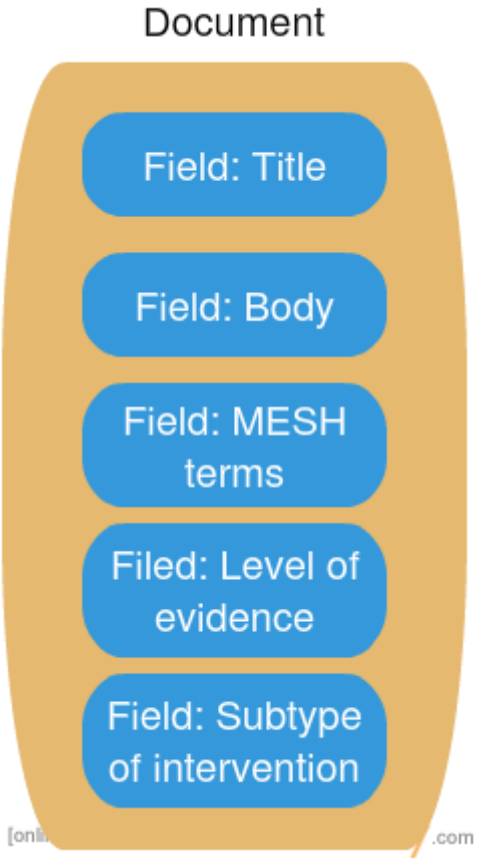

Рис. 9. Модель объекта Document для формирования инвертированного индекса.

Дополнительно для повышения скорости индексации классифицированных аннотаций в модуле построения инвертированного индекса применялась система для распределенных вычислений Spark[27]. Таким образом разработанный модуль индексации позволяет распределённо строить инвертированный индекс. На данный момент поисковой системой проиндексировано 500000 MEDLINE аннотаций с 2006 по 2010 годы.

[27] https://spark.apache.org/

4.2. Модуль обработки запросов

Модуль обработки запросов выполняет два процесса:

1. Предобработка запросов;
2. Булев поиск по запросу в хранилище, содержащем инвертированный индекс.

Процесс предобработки запросов для данного модуля является основным, он выполняет токенизацию запроса, удаление знаков препинания и стоп слов, применяет стемминг.

Процесс булева поиска является первым шагом на этапе ранжирования результатов поиска по уровням доказательности. В результате поиска из хранилища отбираются документы, в которых в поле Title/Body встретилось хотя бы одно слово из запроса. Такие слабые ограничения на поиск позволяют получить большой список документов для выполнения ранжирования.

4.3. Модуль ранжирования по уровням доказательности

Данный модуль реализует алгоритм ранжирования аннотаций по уровням доказательности, выполняя следующие шаги:

1. Аннотации, полученные при помощи булева поиска, собираются в матрицу размера $m \times n$, где m – подтипы медицинских вмешательств, n – уровни доказательности.
2. Внутри каждого элемента матрицы применяется функция tf-idf, вычисляющая оценку релевантности аннотации запросу.
3. Внутри каждого элемента матрицы выполняется ранжирование аннотаций по убыванию оценки релевантности, полученной на 2 шаге.

Приведем пример работы алгоритма ранжирования. Допустим, в результате булева поиска по запросу “breast cancer” мы получили 5 аннотаций со значениями: “Лекарственный препарат” в поле Subtype of intervention и “Рандомизированное открытое” в поле Level of evidence; 7 аннотаций со значениями: “Генетические” в поле Subtype of intervention и

“Нерандомизированные с двойным ослеплением” в поле Level of evidence; 2 аннотации со значениями: “Процедуры” в поле Subtype of intervention и “Мета-анализ” в поле Level of evidence. В результате получим матрицу размера 8×8 (т.е. 8 подтипов медицинских вмешательств и 8 уровней доказательности), в которой буду заполнены только три элемента, а именно: элемент со строкой “Лекарственный препарат” и столбцом “Рандомизированное открытое”, элемент со строкой “Генетические” и столбцом “Нерандомизированные с двойным ослеплением” и элемент со строкой “Процедуры” и столбцом “Мета-анализ”. Далее, внутри каждого элемента вычислим оценку релевантности аннотаций запросу при помощи функцию tf-idf:

$$score(q, d) \quad = \sum_t \ \mathrm{tf}(t \ in \ d) \cdot \ \ \mathrm{idf}(t),$$

где q – запрос, d – аннотация, $t \in q$ – слово в запросе.

В итоге ранжируем аннотации в каждом элементе матрицы по убыванию подсчитанных оценок релевантности.

Данный алгоритм ранжирования позволяет пользователю просмотреть результаты поиска в интересующем его элементе матрицы.

4.4. Модуль извлечения фактов

Данный модуль разрабатывается для решения задачи автоматического аннотирования, извлекая из аннотаций следующих факты:

1. Описание медицинского вмешательства;
2. Название заболевания;
3. Симптомы;
4. Название исследовательского института.

Применение данного модуля позволит упростить медицинским экспертам процесс анализа результатов поиска.

В главе 5 более подробно описаны эксперименты с извлечением названия лекарственного препарата из аннотации.

Глава 5. Проведение экспериментов

5.1. Эксперименты с классификацией аннотаций на “обзорные” и “с вмешательством”

Для начала рассматривались классические линейные алгоритмы классификации Linear SVM, Logistic regression, Naive Bayes из библиотеки sklearn[28] на несбалансированном обучающем множестве. Названия алгоритмов в библиотеке: LinearSVC, MultinomialNB, LogisticRegression. Данные алгоритмы классификации были выбраны из-за того, что они могут обучаться за линейное время с линейной памятью и активно применяются для решения задач, связанных с классификацией тестов.

В качестве вектора признаков для аннотаций использовался вектор слов из словаря, составленного по всему корпусу аннотаций. Вес каждого слова в аннотации оценивался метрикой tf-idf. Гиперпараметры для классификаторов подбирались при помощи метода random layout и метода скользящего контроля на пяти блоках. По результатам случайного поиска были подобраны следующие параметры:

1. LinearSVC: penalty = ‘l1’, loss='squared_hingle', multi_class=’ovr’ , C = 0.5
2. MultinomialNB: alpha=0.4, fit_prior=True;
3. LogisticRegression: penalty = ‘l2’, multi_class=’ovr’ ,C = 0.8 , solver = ‘liblinear’.

Все результаты классификации приведенные, в табл. 10, 11, 12 получены при выполнении метода скользящего-контроля с разбиением на 5- блоков , без решения проблемы не сбалансированности обучаемого множества.

[28] http://scikit-learn.org/

Таблица 10. Результаты Linear SVM

Класс	**Точность**	**Полнота**	**F-мера**
С вмешательством	0.9302	0.6502	0.7651
Обзорные	0.6341	0.7533	0.6885

Таблица 11. Результаты Logistic regression

Класс	**Точность**	**Полнота**	**F-мера**
С вмешательством	0.8422	0.7502	0.7981
Обзорные	0.6421	0.6782	0.6635

Таблица 12. Результаты Naive Bayes

Класс	**Точность**	**Полнота**	**F-мера**
С вмешательством	0.8732	0.6502	0.7531
Обзорные	0.5421	0.6782	0.6435

Учитывая тот факт, что “Обзорных” исследований - 741, а Исследований “с медицинским вмешательством” - 7817, было решено выполнить генерацию повторной выборки (resampling) класса “С вмешательством”, до размеров класса “Обзорные”, применяя технику генерации синтетических объектов SMOTE (Synthetic Minority Over-sampling Technique) [11]. генерации повторных выборок (resampling) большего класса до размеров меньшего защищает обучающее множество от появления дубликатов и выполняет его балансировку.

В результате подбора гиперпараметров для классификаторов на сбалансированных данных были получены следующие результаты:

1. LinearSVC: penalty = ‘l2’, loss='squared_hingle', multi_class=’ovr’ , C = 1.0

2. MultinomialNB: alpha=0.8, fit_prior=True;

3. LogisticRegression: penalty = ‘l1’, multi_class=’ovr’, C = 0.7 , solver = ‘liblinear’.

Все результаты классификации, приведенные в табл. 13, 14, 15 получены при выполнении метода скользящего контроля с разбиением на пять блоков при сбалансированном обучаемом множестве.

Таблица 13. Результаты Linear SVM

Класс	**Точность**	**Полнота**	**F-мера**
С вмешательством	0.8932	0.7533	0.8173
Обзорные	0.8513	0.7611	0.8031

Таблица 14. Результаты Logistic regression

Класс	**Точность**	**Полнота**	**F-мера**
С вмешательством	0.8832	0.7421	0.8060
Обзорные	0.8633	0.7198	0.7850

Таблица 15. Результаты Naive Bayes

Класс	**Точность**	**Полнота**	**F-мера**
С вмешательством	0.8532	0.7021	0.7703
Обзорные	0.7749	0.6933	0.7318

В результате эксперимента можно заметить, что генерации повторных выборок (resampling) улучшило результаты всех алгоритмов классификации, и самый лучший результат показал алгоритм Linear SVM с параметрами:

penalty = ‘l2’, loss='squared_hingle', multi_class=’ovr’ , C = 1.0.

Однако генерации повторных выборок (resampling) с понижением числа объектов в классе “С вмешательством” может привести к недообучению моделей. В связи с этим было решено генерировать синтетические аннотации для класса “Обзорные”, применяя подход, описанный в Главе 2 (пункт 2.3.).

В результате подбора гиперпараметров для классификаторов на сбалансированных данных при помощи синтетических аннотаций были получены следующие результаты:

1. LinearSVC: penalty = ‘l2’, loss='squared_hingle', multi_class=’ovr’ , C = 1.0
2. MultinomialNB: alpha=1.0, fit_prior=True;
3. LogisticRegression: penalty = ‘l2’, multi_class=’ovr’ ,C = 1.0 , solver = ‘liblinear’.

Все результаты классификации, приведенные в табл. 16, 17, 18 получены при выполнении метода скользящего контроля с разбиением на 5-блоков, при сбалансированном обучаемом множестве.

Таблица 16. Результаты Linear SVM

Класс	**Точность**	**Полнота**	**F-мера**
С вмешательством	0.9364	0.7505	0.8332
Обзорные	0.9291	0.7654	0.8394

Таблица 17. Результаты Logistic regression

Класс	**Точность**	**Полнота**	**F-мера**
С вмешательством	0.8832	0.7433	0.8077
Обзорные	0.8913	0.7832	0.8221

Таблица 18. Результаты Naive Bayes

Класс	**Точность**	**Полнота**	**F-мера**
С вмешательством	0.8794	0.7398	0.8035
Обзорные	0.8683	0.7411	0.7996

Можно заметить, как в результате повышения числа аннотаций в классе “Обзорные” за счет генерации синтетических аннотаций выросло качество классификаторов. Полученный результат защищен от недообучения тем, что мы увеличиваем объем класса с меньшим числом объектов до объема класса с большим числом объектов. Благодаря применению метода скользящего контроля с 5-блоками оценки классификаторов не показывают результат переобучения.

5.2. Эксперименты с классификацией по уровням доказательности

На первом этапе было решено сравнить качество AdaBoost с Random Forest и SVM с RBF ядром без выполнения декомпозиции уровней доказательности. Классификаторы были взяты из библиотеки WEKA. В качестве вектора признаков для аннотаций использовался вектор слов из словаря, составленного по всему корпусу аннотаций. Вес каждого слова в аннотации оценивался метрикой tf-idf. Гиперпараметры на несбалансированном обучающем множестве подбирались при помощи метода Random Layout и скользящего контроля с 5-блоками. Были получены следующие гиперпараметры:

для SVM на RBF ядрах:

-S 0 -K 2 -D 3 -G 0.0 -R 0.0 -N 0.5 -M 40.0 -C 1.0 -E 0.001 -P 0.1.

для AdaBoost в сочетании с RandomForest:

-I 100 -K 0 -S 1.

Оценки качества классификации (табл.19, 20) по подтипам медицинского вмешательства были получены с применением метода скользящего контроля с 5-блоками.

Таблица 19. Результаты плоской классификации SVM на RBF ядрах

Класс	**Точность**	**Полнота**	**F-мера**
Рандомизированные с одинарным ослеплением	0.7062	0.6830	0.1245
Рандомизированные с двойным ослеплением	0.7521	0.8333	0.7913
Рандомизированные открытые	0.5342	0.7925	0.6314
Нерандомизрованные с одинарным ослеплением	0	0	0
Нерандомизированные с двойным ослеплением	0	0	0
Нерандомизированные открытые	0.8821	0.1416	0.2502

Таблица 20. Результаты плоской классификации AdaBoost в сочетании с RandomForest

Класс	**Точность**	**Полнота**	**F-мера**
Рандомизированные с одинарным ослеплением	0.8752	0.802	0.1464
Рандомизированные с двойным ослеплением	0.6931	0.8774	0.7741

Рандомизированные открытые	0.5364	0.6863	0.6025
Нерандомизированные с одинарным ослеплением	0	0	0
Нерандомизированные с двойным ослеплением	0	0	0
Нерандомизированные открытые	0.8517	0.1172	0.5773

В результате можно заметить, что из-за малого числа аннотаций в классах “Нерандомизированные с одинарным ослеплением”, “Нерандомизированные с двойным ослеплением” аннотации невозможно распознать. В связи с этим, было решено применить декомпозицию уровней доказательности и сравнить качество классификаторов на отдельных подзадачах:

- классификация по рандомизации;
- классификация по виду ослепления.

При этом дополнительно будет применен метод генерации синтетических аннотаций для балансировки обучающих множеств. Решение балансировки методом генерации синтетических аннотаций было принято исходя из результатов, полученных при классификации аннотаций на “обзорные” и “с вмешательством”.

Поскольку производилась декомпозиция классов, то было решено помимо генерации синтетических аннотаций дополнить обучающее множество вспомогательным. Для формирования вспомогательного множества повторно использовался скрипт, описанный в главе 1, с дополнительной модификацией, позволяющей размечать аннотации, при индексировании которых на сайте clinicaltrials.gov и Isrctn.com указана информация либо только о рандомизации, либо только об ослеплении. В

результате было сформировано вспомогательное обучающее множество (табл. 21), дополнительно используемое при балансировке.

Таблица 21. Вспомогательная коллекция

Класс	**Число аннотаций**
Нерандомизированные	2553
Рандомизированные	4533
С одинарным ослеплением	25
С двойным ослеплением	102
Открытые	83

Для оценки качества классификации аннотаций на рандомизированные и нерандомизированные сравнивались алгоритмы AdaBoost с Random Forest и SVM с RBF ядром. Гиперпараметры подбирались при помощи метода Random Layout и скользящего контроля с 5-блоками. Были получены следующие гиперпараметры:

для SVM на RBF ядрах :

-S 0 -K 2 -D 3 -G 0.0 -R 0.0 -N 0.5 -M 40.0 -C 0.8 -E 0.001 -P 0.1.

для AdaBoost в сочетании с RandomForest:

-I 100 -K 0 -S 3.

Оценки качества классификации аннотаций (табл. 22, 23) на рандомизированные и нерандомизированные были получены с применением метода скользящего контроля с 5-блоками. Дополнительно для балансировки использовались синтетические аннотации и аннотации из вспомогательного множетсва.

Таблица 22. Результаты классификации на рандромизированные и нерандомизированные AdaBoost в сочетании с RandomForest

Точность	**Полнота**	**F-мера**	**Класс**
0.8524	0.8121	0.8312	Рандомизированные
0.8131	0.8533	0.8334	Нерандомизированные

Таблица 23. Результаты классификации на рандромизированные и нерандомизированные SVM с RBF ядром

Точность	**Полнота**	**F-мера**	**Класс**
0.8792	0.8651	0.8721	Рандомизированные
0.8622	0.8763	0.8694	Нерандомизированные

В результате эксперимента можно заметить, что лучший результат классификации достигается с применением нелинейной разделяющей поверхности в алгоритме SVM. Гиперпараметры алгоритма SVM с RBF ядром:

-S 0 -K 2 -D 3 -G 0.0 -R 0.0 -N 0.5 -M 40.0 -C 0.8 -E 0.001 -P 0.1

Для оценки качества классификации аннотаций по видам ослеплений сравнивались алгоритмы AdaBoost с Random Forest и SVM с RBF ядром. Гиперпараметры подбирались при помощи метода Random Layout и скользящего контроля с 5-блоками. Были получены следующие гиперпараметры для SVM с RBF ядром:

-S 0 -K 2 -D 3 -G 0.0 -R 1.2 -N 0.5 -M 40.0 -C 1 -E 0.001 -P 0.1.

Для AdaBoost в сочетании с RandomForest:

-I 200 -K 2 -S 5.

Оценки качества классификации аннотаций (табл. 24, 25) на рандомизированные и нерандомизированные были получены с применением метода скользящего контроля с 5-блоками. Дополнительно для балансировки

использовались синтетические аннотации и аннотации из вспомогательного множества.

Таблица 24. Результаты классификации аннотаций по видам ослеплений SVM с RBF ядром

Точность	**Полнота**	**F-мера**	**Класс**
0.8912	0.7561	0.8183	С двойным ослеплением
0.5935	0.9202	0.7213	Открытые
0.6253	0.1271	0.2115	С одинарным ослеплением

Таблица 25. Результаты классификации аннотаций по видам ослеплений AdaBoost в сочетании с Random Forest

Точность	**Полнота**	**F-мера**	**Класс**
0.9364	0.7505	0.8332	С двойным ослеплением
0.9291	0.7654	0.8394	Открытые
0.6714	0.9362	0.7820	С одинарным ослеплением

В результате эксперимента можно заметить, что нелинейная разделяющая поверхность SVM плохо разделяет данные классы, поскольку SVM изначально является не мульти-классовым классификатором, и в данной задаче он использует модификацию One-vs-Rest. В свою очередь, алгоритм AdaBoost в сочетании с Random Forest показывает лучший результат,

поскольку выполняет задачу мульти-классификации без дополнительных модификаций.

Гиперпараметры AdaBoost в сочетании с Random Forest:

-I 200 -K 2 -S 5.

В результате слияния лучших результатов классификации для задач 1 и 2 получим следующие оценки качества классификации по уровням доказательности (табл. 26):

Таблица 26. Результаты классификации при декомпозиции уровней доказательности

Класс	**Точность**	**Полнота**	**F-мера**
Рандомизированные с двойным ослеплением	0.9264	0.7505	0.8232
Рандомизированные с одинарным ослеплением	0.8634	0.7211	0.7858
Рандомизированные открытые	0.9122	0.8264	0.8671
Нерандомизрованные с одинарным ослеплением	0.9234	0.7513	0.8335
Нерандомизированные с двойным ослеплением	0.8732	0.7531	0.8617
Нерандомизированные открытые	0.8935	0.7216	0.7984

Можно заметить, что декомпозиция уровней доказательности улучшает качество классификации аннотаций за счет выполнения балансировки обучающих множеств и выбора лучшего алгоритма классификации для каждой подзадачи отдельно.

5.3. Классификация по подтипам медицинских вмешательств

В связи с тем, что выполнять декомпозицию подтипов медицинских вмешательств было не возможно из-за природы классов, входящих в Список 2, было решено сравнить следующие алгоритмы классификации: Linear SVM, Logistic regression, Naive Bayes, дополнительно применяя алгоритм LSI для выбора признаков. Все алгоритмы были взяты из библиотеки sklearn. Названия алгоритмов в библиотеке: LinearSVC, MultinomialNB, LogisticRegression. Данные алгоритмы классификации были выбраны из-за того, что для задачи классификации аннотаций на “обзорные” и “с вмешательством” они показали лучший результат.

В качестве вектора признаков для аннотаций использовался вектор слов из словаря, составленного по всему корпусу аннотаций. Вес каждого слова в аннотации оценивалсяась метрикой tf-idf. Гиперпараметры для классификаторов подбирались при помощи метода random layout и метода скользящего контроля на пяти 5-блоках. Гиперпараметры:

1. LinearSVC: penalty = ‘l1’, loss='squared_hingle', multi_class=’ovr’ , C = 0.5
2. MultinomialNB: alpha=0.4, fit_prior=True;
3. LogisticRegression: penalty = ‘l2’, multi_class=’ovr’ ,C = 0.8 , solver = ‘liblinear’.

Оценки качества классификации аннотаций (табл. 27, 28) по подтипам медицинских вмешательств были получены с применением метода скользящего контроля с 5-блоками и алгоритмом LSA для отбора признаков. При помощи LSA были отобраны 210 лучших признаков. Число лучших признаков подбиралось при помощи метода random layout и метода скользящего контроля на пяти 5-блоках.

Таблица 27. Результаты классификации подтипов медицинских вмешательств с применением алгоритма LSA (лучшие 210 признаков)

Класс	**Точность**			**Полнота**			**F-мера**		
	Naive Bayes	Linear SVC	Logistic Regression	Naive Bayes	Linear SVC	Logistic Regression	Naive Bayes	Linear SVC	Logistic Regression
Поведенческие	0.7943	0.7316	0.7134	0.2241	0.8133	0.7644	0.3524	0.7636	0.7415
Биологические препараты	1	0.8812	0.9614	0.0314	0.7452	0.6512	0.0612	0.80	0.7726
Устройства	0	0.5632	0.7531	0.7511	0.3927	0.1287	0	0.4684	0.2156
Пищевые добавки	0	0.57	0.6213	0	0.6782	0.2893	0	0.6277	0.3935
Лекарственные препараты	0.50	0.8133	0.6567	1	0.9257	0.97	0.66	0.87	0.7814
Генетические	0	0.3442	0.5178	0	0.2216	0.2245	0	0.2715	0.19
Процедуры	0	0.6136	0.6417	0	0.4338	0.1535	0	0.5014	0.2515
Радиация	0	1	0	0	0.3316	0	0	0.5944	0

Таблица 28. Сравнение результатов классификации с LSA

Классификатор	**Точность**		**Полнота**		**F-мера**	
	микро	макро	микро	макро	микро	макро
Naive Bayes	0.5096	0.2858	0.8076	0.1559	0.6249	0.2017
Linear SVC	0.6249	0.6876	0.7683	0.5643	0.6892	0.6199
Maximum Entropy	0.6428	0.6033	0.7535	0.3825	0.6938	0.4161

На втором этапе сравнивались классификаторы без применения LSA. Гиперпараметры для классификаторов подбирались при помощи метода

random layout и метода скользящего-контроля на пяти 5-блоках. Гиперпараметры:

1. LinearSVC: penalty = ‘l2’, loss='squared_hingle', multi_class=’ovr’ , C = 1
2. MultinomialNB: alpha=0.8, fit_prior=True;
3. LogisticRegression: penalty = ‘l1’, multi_class=’ovr’ ,C = 1 , solver = ‘liblinear’.

Оценки качества классификации аннотаций (табл. 29, 30) по подтипам медицинских вмешательств были получены с применением метода скользящего контроля с 5-блоками без применения LSA.

Таблица 29 Результаты классификации подтипов медицинских вмешательств без LSA

Класс	**Точность**			**Полнота**			**F-мера**		
	Naive Bayes	Linear SVC	Logistic Regression	Naive Bayes	Linear SVC	Logistic Regression	Naive Bayes	Linear SVC	Logistic Regression
Поведенческие	0.7781	0.7366	0.7051	0.4956	0.7814	0.7612	0.60	0.7589	0.7316
Биологические препараты	1	0.89	0.96	0.2417	0.7245	0.65	0.3865	0.8	0.7715
Устройства	0	0.6274	0.7145	0	0.4152	0.1231	0	0.49	0.1867
Пищевые добавки	0	0.5812	0.58	0	0.6317	0.2466	0	0.60	0.3415
Лекарственные препараты	0.5312	0.7988	0.6512	0.9981	0.94	0.97	0.69	0.8628	0.7826
Генетические	1	0.34	0.50	1	0.2278	0.0917	0.0318	0.2758	0.1535
Процедуры	0	0.6479	0.65	0	0.42	0.1452	0	0.5188	0.2374
Радиация	0	1	0	0	0.3345	0	0	0.5067	0

Таблица 30. Сравнение результатов классификации без LSA

Классификатор	**Точность**		**Полнота**		**F-мера**	
	микро	макро	микро	макро	микро	макро
Multinomial Naive Bayes	0.5577	0.4128	0.8687	0.2162	0.6793	0.28378%
Linear SVC	0.6491	0.7005	0.7913	0.5563	0.7132	0.6201%
Maximum Entropy	0.6566	0.5934	0.7664	0.3694	0.7073	0.4553%

В результате проведенных экспериментов можно заметить, что лучшую оценку качества показал алгоритм Linear SVC без применения LSA.

Гиперпараметры алгоритма:

penalty = ‘l2’, loss='squared_hingle', multi_class=’ovr’ , C = 1

5. 4. Эксперименты с извлечением фактов

В данном эксперименте сравнивались два подхода для решения задачи извлечения наименования лекарственного препарата из медицинского исследования.

В рамках первого подхода, основанного на поисках фразовых шаблонов [7], существует возможность оценить вероятность принадлежности слова названию лекарственного препарата. В качестве шаблона можно рассматривать следующий пример <X compared with Y> [7]. Зная шаблон, можно извлечь из текста все возможные значения <X,Y>. В случае если имеется небольшое количество шаблонов, их можно пополнить, используя алгоритм bootstrapping [48].

В данном эксперименте не было исходного списка шаблонов, в связи с чем для их поиска было решено построить языковую модель, основанную на *n*-граммах [48]. Построенные шаблоны были оценены экспертами. Языковая модель (биграммная и триграммная) строилась при помощи Lucene без

выполнения стемминга и удаления стоп-слов. Результаты построения языковых моделей приведены в таблице 31.

Таблица 31. Топ 10 биграмм и триграмм

Биграммы	**Вероятность *n*-граммы в языковой модели**	**Триграммы**	**Вероятность *n*-граммы в языковой модели**
in the	0.0055	in patients with	0.0013
patients with	0.0035	95 confidence interval	0.0005
of the	0.0028	the placebo group	0.0004
in patients	0.0017	the treatment of	0.0004
associated with	0.0015	were randomized to	0.0004
compared with	0.0013	was associated with	0.0004
for the	0.0013	in the placebo	0.0003
and the	0.0011	randomly assigned to	0.0003
of patients	0.0010	were randomly assigned	0.0003
the primary	0.0009	the effect of	0.0003

Полученные модели анализировались экспертами из области медицины. Результатом анализа явилось то, что шаблоны, полученные в результате данного подхода, не могут быть использованы для извлечения названий лекарственного препарата. Шаблоны с максимальной вероятностью встречи в языковой модели представляют последовательность стоп-слов, что уменьшает вероятность найти X,Y соответствующие названиям лекарственных препаратов.

В рамках второго подхода было предложено использовать частный случай марковских случайных полей – модель CRF, которая была успешно применена в задаче [49] извлечения именованных сущностей для коллекции

медицинских текстов. Для применения CRF было сформировано обучающее множество (глава 2 пункт 2.2.).

Для обучения и оценки CRF применялся метод скользящего контроля с 5-блоками. Дополнительно при методе скользящего контроля рассматривались два случая:

1. Замена названий “Лекарственных препаратов” в тестовом разбиении на случайные числа;
2. Без замены названий “Лекарственных препаратов” в тестовом разбиении.

Рассмотрение таких случаев позволяет оценить качество модели, не опасаясь того, что алгоритм переобучится и будет работать как простой поиск по ключевым словам без применения *n*-граммных лексико-грамматических признаков. Результаты экспериментов с лучшим набором признаков приведены в табл. 32.

Для работы с алгоритмом CRF использовалась библиотека CRF++. Ссылки на python скрипты и более детальное описание подготовки обучающего множества, а так же обучение и оценка модели CRF с применением скользящего среднего выложены на ресурсах GitHub[29] и Bitbucket[30].

Таблица 32. Результаты экспериментов извлечения наименования лекарственного препарата

Эксперименты	**1**	**2**
Скрываем название препарата	FALSE	TRUE
Количество слов в тестовой выборке	140729	140729
Количество слов в обучающей выборке	469096	469096
Точность	99.45%	96.31%

[29] https://gist.github.com/KamalovMikhail/d98bbec36d9363f83fbf1f3270cdaa17

[30] https://github.com/KamalovMikhail/search_engine/tree/master/learning_set/src/stanford_parser

Полнота	86.49%	4.02%
Количество лекарственных препаратов в тестовой выборке	2619	2834
Количество лекарственных препаратов в обучающей выборке	10479	10264
Количество похожих лекарственных препаратов в обучающей и тестовой выборках	2342	2401

В результатам эксперимента можно заметить, что модель CRF переобучается на словах. Это явно выражается в уменьшении значения полноты при замене названий лекарственных препаратов в тестовом разбиении на случайные числа и в количестве одинаковых лекарственных препаратов в обучающих и тестовых разбиениях.

В дальнейшем предлагается использовать модель максимальной энтропии, а также word2vec[31] для построения семантического описания, позволяющего расширить набор признаков для модели CRF и уйти от непосредственного учета слов, тем самым защищаясь от переобучения.

[31] http://deeplearning4j.org/word2vec

Заключение

Разработанная поисковая система основана на комбинации классификаторов, определяющих уровень доказательности аннотации и подтип медицинского вмешательства. Предсказания, полученные от классификаторов, позволяют представлять результаты поиска в матричной форме. При этом в строке, соответствующей определенному подтипу медицинского вмешательства, и в столбце, соответствующем уровню доказательности, расположены аннотации, отсортированные в порядке убывания оценки релевантности аннотации запросу. Оценка релевантности найденной аннотации запросу считается при помощи функции tf-idf. На данный момент реализованная поисковая система, ранжирующая MEDLINE аннотации по уровням доказательности тестируется медицинскими экспертами.

В процессе разработки данной системы были опубликованы результаты экспериментов, в том числе классификации MEDLINE аннотаций по подтипам медицинских вмешательств [49] и сравнение стандартных алгоритмов кластеризации с применением алгоритмов выбора признаков для задачи кластеризации MEDLINE аннотаций по подтипам медицинских вмешательств [19, 50].

В дальнейшем планируется:

1. улучшить качество работы модуля фильтрации;
2. повысить качество работы модуля классификации;
3. учитывать в алгоритме ранжирования ошибку классификаторов;
4. учитывать в алгоритме ранжирования критерии оценок GRADE;
5. улучшить качество извлечения фактов.

Список литературы

1. Guyatt, G., Cairns, J., Churchill, D., Cook, D., Haynes, B., Hirsh, J., Sackett, D. (1992). Evidence-based medicine: a new approach to teaching the practice of medicine. *Jama*, *268*(17), 2420-2425.
2. Wong, C. K., Ho, C. Y., Li, E. K., & Lam, C. W. K. (2000). Elevation of proinflammatory cytokine (IL-18, IL-17, IL-12) and Th2 cytokine (IL-4) concentrations in patients with systemic lupus erythematosus. *Lupus*, *9*(8), 589-593.
3. G. Guyatt,G. Vist, Y. Falck-Ytter, R. Kunz, N. Magrini ,H. Schunemann for the GRADE* working group. "An emerging consensus on grading recommendations?," (Editorial). ACP J Club, 2006, Jan-Feb;144(1):A08,PMID: 17216711.
4. G. Guyatt,G. Vist, Y. Falck-Ytter, R. Kunz, N. Magrini ,H. Schunemann for the GRADE* working group."GRADE guidelines: 1. Introduction-GRADE evidence profiles and summary of findings tables,"Journal of Clinical Epidemiology, 2011, vol. 64, pp. 383-394, doi:10.1016/j.jclinepi.2010.04.026.
5. Fyfe, T. (2007). Turning Research Into Practice (TRIP). *Journal of the Medical Library Association*, *95*(2), 215.
6. Ohta, T., Tsuruoka, Y., Takeuchi, J., Kim, J. D., Miyao, Y., Yakushiji, A., ... & Hara, T. (2006, July). An intelligent search engine and GUI-based efficient MEDLINE search tool based on deep syntactic parsing. In *Proceedings of the COLING/ACL on Interactive presentation sessions* (pp. 17-20). Association for Computational Linguistics.
7. Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval* (Vol. 1, No. 1, p. 496). Cambridge: Cambridge university press.
8. Büttcher, S. Content-and-Structure Queries in an XML-based Information Retrieval System.

9. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16:341–378, 2002.
10. Blei, David M.; Ng, Andrew Y.; Jordan, Michael I (January 2003). Lafferty, John, ed. "Latent Dirichlet Allocation". Journal of Machine Learning Research 3 (4–5): pp. 993–1022. doi:10.1162/jmlr.2003.3.4-5.993
11. Sutton, C., & McCallum, A. (2011). An introduction to conditional random fields. Machine Learning, 4(4), 267-373.
12. S. Choi, B. Ryu, S. Yoo, and J. Choi. Combining relevancy and methodological quality into a single ranking for evidence-based medicine. Information Sciences: an International Journal, 214:76–90, December 2012.
13. S. E. Robertson and S. Walker. Okapi/keenbow at trec-8. In Proceedings The Eighth REtrieval Conference (TREC-8), pages 151–162, November 1999.
14. H. Kilicoglu, D. Demner-Fushman, T. Rindflesch, N. Wilczynski, and R. Haynes. Towards automatic recognition of scientifically rigorous clinical research evidence. Journal of the American Medical Informatics Association, 16(1):25–31, October 2009.
15. K. Veningston, R. Shanmugalakshmi, "Information Retrieval by Document Re-ranking using Term Association Graph," Proceedings of the 2014 International Conference on Interdisciplinary Advances in Applied Computing, New York, USA, 2014 , vol. 8, Article No. 21., doi:10.1145/2660859.2660927
16. A. M. Cohen, C. E. Adams, J. M. Davis, C. Yu, P. S. Yu, W. Meng, L. Duggan, M. McDonagh, and N. R. Smalheiser. Evidence-based medicine, the essentialrole of systematic reviews, and the need for automated text mining tools. In Proceedings of the 1st ACM International Health Informatics Symposium (IHI'10), pages 376–380. ACM, November 2010.
17. K. S. Khan, R. Kunz, J. Kleijnen, and G. Antes. Fivesteps to conducting a systematic review. Journal of the Royal Society of Medicine, 96(3):118–121, March 2003.

18. B. R¨ohrig, J.-B. du Prel, D. Wachtlin, and M. Blettner. Types of study in medical research. ¨Deutsches Arzteblatt International Journal, 106(15):262–268, April 2009.
19. P. Davis-Desmond and D. Moll´a. Detection of evidence in clinical research papers. In Proceedings of the Fifth Australasian Workshop on Health Informatics and Knowledge Management (HIKM ’12), volume 129, pages 13–20. ACM, 2012.
20. K. McKibbon, N. Wilczynski, R. Haynes, and T. Hedges. Retrieving randomized controlled trials from medline: a comparison of 38 published search filters. Health Information and Libraries Journal, 26(3):187–202, September 2009.
21. I. Yoo, X. Hu “A comprehensive comparison study of document clustering for a biomedical digital library MEDLINE,” Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries, New York, USA, 2006, pp. 220-229, doi:10.1145/1141753.1141802.
22. Dobrynin, V., Balykina, Y., & Kamalov, M. (2015, October). Analysis of standard clustering algorithms for grouping MEDLINE abstracts into evidence-based medicine intervention categories. In " Stability and Control Processes" in Memory of VI Zubov (SCP), 2015 International Conference (pp. 555-557). IEEE.
23. V. Dobrynin, D. Patterson, M. Galushka, N. Rooney, “SOPHIA: An In-teractive Cluster Based Retrieval System for the OHSUMED collection,” in IEEE Trans. on Information Technology for Biomedicine, 2005, vol. 9, pp. 256-265, PMID: 16138542.
24. I. Yoo, X. Hu, Il-Y. Song, “Integration of semantic-based bipartite graph representation and mutual refinement strategy for biomedical literature clustering,” Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining , New York, USA, 2006, pp. 791-796, doi: 10.1145/1150402.1150505.
25. D. Demner-Fushman, J. Lin, “Answer extraction, semantic clustering, and extractive summarization for clinical question answering,” Proceedings of the

21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics , Stroudsburg, USA, 2006, pp. 841-848, doi: 10.3115/1220175.1220281

26. Shultz, M. (2007). Comparing test searches in PubMed and Google Scholar. JOURNAL-MEDICAL LIBRARY ASSOCIATION, 95(4), 442.
27. Anders, M. E., & Evans, D. P. (2010). Comparison of PubMed and Google Scholar literature searches. Respiratory care, 55(5), 578-583.
28. Srinivasan, P. (1996). Optimal document-indexing vocabulary for MEDLINE. Information Processing & Management, 32(5), 503-514.
29. Trieschnigg, D., Hiemstra, D., de Jong, F., & Kraaij, W. (2010, October). A cross-lingual framework for monolingual biomedical information retrieval. In Proceedings of the 19th ACM international conference on Information and knowledge management (pp. 169-178). ACM.
30. Limsopatham, N., Macdonald, C., & Ounis, I. (2013, July). Learning to combine representations for medical records search. In Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval (pp. 833-836). ACM.
31. Hara, K., & Matsumoto, Y. (2007). Extracting clinical trial design information from MEDLINE abstracts. New Generation Computing, 25(3), 263-275.
32. Fiszman, M., Demner-Fushman, D., Kilicoglu, H., & Rindflesch, T. C. (2009). Automatic summarization of MEDLINE citations for evidence-based medical treatment: A topic-oriented evaluation. Journal of biomedical informatics, 42(5), 801-813.
33. S. Kaneko, A. Hayashi, N. Suematsu, K. Iwata, "Hierarchical hidden conditional random fields for information extraction," Proceedings of the 5th international conference on Learning and Intelligent Optimization, Springer-Verlag Berlin, Heidelberg, 2011, vol. 12, pp. 191-202, doi: 10.1007/978-3-642-25566-3_14.
34. A. Hliaoutakis, K. Zervanou, E. G.M. Petrakis, E. E. Milios, "Automatic document indexing in large medical collections," Proceedings of the

international workshop on Healthcare information and knowledge management, New York, USA, 2006, vol. 8, pp. 1-8, doi:10.1145/1183568.1183570.

35. Cortes, C.; Vapnik, V. (1995). "Support-vector networks". Machine Learning 20 (3): 273. doi:10.1007/BF00994018.
36. Hsu, Chih-Wei; and Lin, Chih-Jen (2002). "A Comparison of Methods for Multiclass Support Vector Machines". *IEEE Transactions on Neural Networks*.
37. Boser, B. E.; Guyon, I. M.; Vapnik, V. N. (1992). "A training algorithm for optimal margin classifiers". *Proceedings of the fifth annual workshop on Computational learning theory – COLT '92*. p. 144. doi:10.1145/130385.130401. ISBN 089791497X.
38. Manevitz, L. M., & Yousef, M. (2002). One-class SVMs for document classification. the Journal of machine Learning research, 2, 139-154.
39. Russell, Stuart; Norvig, Peter (2003) [1995]. Artificial Intelligence: A Modern Approach (2nd ed.). Prentice Hall. ISBN 978-0137903955.
40. Brzezinski, J. R. (2000). Logistic regression for classification of text documents. DePaul University, School of Computer Science, Telecommunications, and Information Systems.
41. H. He and Y. Ma. Imbalanced Learning: Foundations, Algorithms, and Applications. Wiley Publishing, 2013.
42. J. J. Rodr´ıguez, J. F. D´ıez-Pastor, and C. Garc´ıa-Osorio. Ensembles of decision trees for imbalanced data. In Proceedings of the 10th international conference on Multiple classifier systems (MCS'11), pages 76–85. Springer-Verlag, November 2011.
43. Ho, Tin Kam (1995). Random Decision Forests (PDF). Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC, 14–16 August 1995. pp. 278–282.
44. Breiman, Leo (2001). "Random Forests". Machine Learning 45 (1): 5–32. doi:10.1023/A:1010933404324.
45. Freund, Y., Schapire, R., & Abe, N. (1999). A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, *14*(771-780), 1612.

46. Bergstra, J. and Bengio, Y., Random search for hyper-parameter optimization, The Journal of Machine Learning Research (2012)
47. Susan T. Dumais (2005). "Latent Semantic Analysis". Annual Review of Information Science and Technology 38: 188. doi:10.1002/aris.1440380105.
48. Martin, J. H., & Jurafsky, D. (2000). Speech and language processing. International Edition.
49. Krallinger, M., Leitner, F., Rabal, O., Vazquez, M., Oyarzabal, J., & Valencia, A. (2015). CHEMDNER: The drugs and chemical names extraction challenge. J. Cheminformatics, 7(S-1), S1.
50. Dobrynin, V., Balykina, J., Kamalov, M., Kolbin, A., Verbitskaya, E., & Kasimova, M. (2015, September). The data retrieval optimization from the perspective of evidence-based medicine. In Computer Science and Information Systems (FedCSIS), 2015 Federated Conference on (pp. 323-328). IEEE.
51. Kamalov, M., Dobrynin, V., Balykina, J., Kolbin, A., Verbitskaya, E., & Kasimova, M. (2015). Improving data retrieval quality: Evidence based medicine perspective. International Journal of Risk & Safety in Medicine, 27(s1), S106-S107.